Topology Optimization for Convection Problems

Alexandersen, Joe

Publication date:
2011

Document Version
Publisher's PDF, also known as Version of record

Topology Optimization for Convection Problems

Topologioptimering for konvektionsproblemer

B.Eng. Exam Project - Diplom Afgangsprojekt

Author: Joe Alexandersen, s072713

Signature: _____

Supervisor: Professor Dr.Techn. Ole Sigmund

DTU Mekanik

Abstract

This report deals with the topology optimization of convection problems. That is, the aim of the project is to develop, implement and examine topology optimization of purely thermal and coupled thermomechanical problems, when the design-dependent effects of convection are taken into consideration. This is done by the use of a self-programmed FORTRAN-code, which builds on an existing 2D-plane thermomechanical finite element code implementing during the course '41525 FEM-Heavy'. The topology optimization features have been implemented from scratch, and allows the program to optimize elastostatic mechanical problems, steady-state thermal problems and coupled thermomechanical problems, for a range of objective functions subjected to a volume-fraction constraint. The programme utilises the GCMMA algorithm, by K. Svanberg, to optimize the given problems. Design-dependent side convection has been formulated and implemented, where the convection is interpolated into the design-domain and applied upon the boundaries of the design using a density-based interpolation-function. The implementation has been tested for both pure thermal analysis, along with thermal and thermomechanical optimization. The solutions to the test problems show interesting results, that underline the significance of including convection and the design-dependent effects thereof.

Resumé

Denne rapport omhandler topologioptimering for konvektionsproblemer. Målet med projektet er altså at udvikle, implementere og undersøge topologioptimering af rent termiske og koblede termomekaniske problemer, når de design-afhængige effekter af konvektion inkluderes. Dette er gjort ved brug af et FORTRAN-program, der bygger på en 2D-plant termomekanisk finite element kode, implementeret i kurset '41525 FEM-Heavy'. Topologioptimeringsfunktionerne er blevet implementeret fra bunden og giver mulighed for at optimere elastostatiske mekaniske problemer, 'steady-state' termiske problemer og koblede termomekaniske problemer. Disse problemer kan optimeres mht. en række objektfunktioner under en påtvunget volumen-begrænsning. Programmet gør brug af GCMMA algoritmen, skrevet af K. Svanberg. Design-afhængigt side konvektion er blevet formuleret og implementeret, hvor konvektionen bliver interpoleret ind i design-domænet og påført grænselinjen af designet, ved brug af en densitet-baseret interpolationsfunktion. Implementeringen er blevet testet for både rent termisk analyse, såvel som termisk og termomekanisk optimering. De opnåede løsninger til test problemerne viser interessante resulter, der understreger vigtigheden af at, inkludere konvektion og de design-afhængige effekter heraf.

Foreword

This 'Bachelor of Engineering - Mechanical Engineering' ('Diplomingeniør - Maskin') exam project has been carried out following the specified guidelines at the Technical University of Denmark (DTU). The project-supervisor has been Professor Dr.Techn. Ole Sigmund. The project counts for 20 ECTS points and has been completed during the 7th and last semester of the bachelor degree, over the period from the 30th of August 2010 to the 21st of January 2011.

I would like to thank Ole Sigmund for being a great supervisor, with good advice and helpful guidelines at times of need. Ole, along with Jakob S. Jensen, also delivered the extremely inspiring teaching and presentations throughout the course '41525 FEM-Heavy', that stimulated me to pursue a deeper understanding of numerical modelling and optimization. I would also like to thank Anders V. Hansen and Henrik Andersson (MAN Diesel & Turbo), along with Henrik B. Clausen (formerly MAN Diesel & Turbo), for initiating an incredibly interesting project, after an extremely pleasant and giving internship. My parents and brother; Gitte, Soren and Mike Alexandersen, should be thanked for keeping me sane and helping me to relax for a few weeks over the December holiday season. My grandmother, Kirsten Hansen, also deserves thanks for providing me with hot food and comfort, whenever I found time to visit her during the busy times. Lastly, but not least, my friends and office-mates should be thanked for making the troublesome and busy times enjoyable.

Contents

Chapter 1

Introduction

The aim of the project is to develop, implement and examine topology optimization of purely thermal and coupled thermomechanical problems, when the design-dependent effects of convection are taken into consideration. This will be done by the use of a FORTRAN programme, that builds on a thermomechanical Finite Element [FE] code implemented during the course '41525 FEM-Heavy'. The first part of the project is to implement basic topology optimization features for static mechanical, steady-state heat transfer and coupled analysis in the code and to verify these. Thereafter, the current developments within topology optimization relevant to design-dependent convection is examined, and a design-dependent convection formulation is then implemented in the FORTRAN programme. Both "top and bottom" and "side" convection will be worked with (these terms are described in chapter 4). An analytical study will be carried out to check the accuracy of the implemented features, and after this has been verified, two test problems will be used to show the effects of design-dependent convection. The first purely thermal and the second a thermomechanical coupled problem. The original goal was to finally solve a realistic industrial problem in cooperation with MAN Diesel & Turbo, where the problem of topology optimization wrt. cooling and strength of a piston would be looked at. But unfortunately, the complexity of the problem at hand did not allow the time to do this.

Topology optimization is where an algorithm is used to find the optimal shape and distribution of material, for given parameters and restrictions. It is based on the classical mathematical discipline of optimization working together with numerical models, such as Finite Element Modelling [FEM], and is, therefore, an incredibly powerful tool. It has its roots in classic shape and size optimization, where topology optimization solves both problems at once. Bendsøe and Kikuchi (together and individually) [1, 2, 3] pioneered the optimization of shape and size as a material distribution problem in the late 80's, and also introduced the homogenization method along

with the penalised variable density approach known as SIMP (see section 2.2). Topology optimization started out as a technology mostly used within structural optimization, but has since been extended to cover things such as compliant mechanisms [5], MicroElectroMechanical Systems (MEMS) [6], thermoelastic analysis [7] and a wide range of other problem from electromagnetics to fluid flows. For more details, see the comprehensive text on topology optimization and its many uses by Bendsøe and Sigmund [24].

Topology optimization has long been a tool used in industries where saving weight is very important, like the automotive and aeronautical industries. Commercial optimization software is developing very fast and becoming better and more user-friendly, but the use of topology optimization is unfortunately not as widespread as it could be. However, with the expanding capabilities and the rapid progress within topology optimization these days, this very powerful tool will hopefully soon be used in a wide range of industries.

Motivation for this project

The motivation for this project came from MAN Diesel & Turbo, where the author spent half-a-year doing an internship, who wanted to optimize the cooling of their pistons subject to several strength and relative displacement constraints. As convection plays a huge role in the cooling process and this was not presently implemented in the commercial software available on the market, the project was defined with a theoretical basis in investigating and implementing design-dependent convection. It quickly turned out that is a relatively unexplored area of topology optimization and the development of this, therefore, became very interesting.

Part I

Theory

Chapter 2

General topology optimization

This chapter covers the theory of general topology optimization, from the basics to some of the problems often encountered and the solution methods used.

2.1 The basics of topology optimization

The problems to be solved using topology optimization, can be written in the following standard form of an optimization problem:

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0\,,\; i = 1, \ldots, m \\ & x_{\min} \leq x_j \leq x_{\max}\,,\; j = 1, \ldots, n \end{aligned} \tag{2.1}$$

where f_0 is the objective function to be minimized, f_i is the m-number of constraint functions and $\mathbf{x}$ is a vector containing the n-number of design variables.

As topology optimization was first used within static mechanics, the equations and examples in this chapter will be confined to this area, except for section 2.5. The objective function is what the optimization algorithm is to minimise and is throughout this project evaluated using Finite Element Analysis [FEA]. When dealing with static mechanical analysis, the objective function, f_0, is most often defined as the compliance of the structure. As the compliance is analogous to the inverse of the stiffness; minimising the compliance is the same as maximising the stiffness. The objective function can in such cases be written as:

$$f_0(\mathbf{x}) = \mathbf{f}^T\mathbf{u} \tag{2.2}$$

$$= \mathbf{u}^T\mathbf{K}(\mathbf{x})\,\mathbf{u} \tag{2.3}$$

where $\mathbf{f}$ and $\mathbf{u}$ are the load and displacement vectors, respectively, from the finite element equilibrium equation for static mechanics:

$$\mathbf{K}(\mathbf{x})\mathbf{u} = \mathbf{f} \tag{2.4}$$

where $\mathbf{K}$ is the stiffness matrix as a function of the design variables, $\mathbf{x}$, as defined in section 2.2.

The constraint functions, f_i, can be anything from nodal displacements to the used volume of the design domain. Most often the weight of the solution is very important in practice, so a volume or volume fraction constraint is used.

During the optimization procedure, the design variable needs to be updated between the iterations to produce a more favourable design wrt. the objective function and the satisfaction of constraint functions. There are several methods that can be used to update the design variable, and a few select of these will be covered in section 2.3.

2.2 Design parametrisation

When topology optimization is formulated as a material distribution problem, the actual problem becomes discrete. The desirable solution is a solution where there either exists or does not exist material, at all points in the continuum. As this report is limited within the topology optimization of structures, a measure of the relative density of material, ρ_e, for each element in the FE discretisation is used as the design variables, $\mathbf{x}$. To ease numerical calculations, the design variables are approximated using continuous variables:

$$0 \leq \rho_e \leq \rho_{\max} \leq 1,\ e = 1, \ldots, n_e \tag{2.5}$$

$\rho_{\max}$ is most often set to 1, so as to allow the design variables to take the value of "fully solid" material - this will be the case throughout this report. The often used volume constraint can now be expressed as follows:

$$f_1(\boldsymbol{\rho}) = \frac{\sum_{e=1}^{n_e}(\rho_e V_e)}{V^*} - 1 \leq 0 \tag{2.6}$$

where ρ_e and V_e is the density and volume of element e respectively, n_e is the number of elements and V^* is the imposed volume constraint.

To avoid areas of intermediate density and to push the continuous variable towards a solution with only densities of 0 and 1 (from now on referred to as a 0-1 solution or design), some sort of penalisation needs to be imposed on intermediate densities. For structural optimization of isotropic materials, the SIMP method is most often used.

SIMP

SIMP stands for Solid Isotropic Material with Penalisation, and is used to interpolate the material parameters as a function of the density. The penalisation is in the form of raising the design variable to the power p, so as to penalise intermediate densities and push the solution towards a 0-1 design. By raising the design variables to a power larger than 1, intermediate values between 0 and 1 will get smaller and, therefore, contribute less to the total stiffness. Poisson's ratio is assumed to be independent of the density and the Young's modulus is parametrised to be dependent of the variable density using the following equation:

$$E(\rho_e) = (\mu_{\min} + (1 - \mu_{\min})\, {\rho_e}^p)\, E^0 \tag{2.7}$$

where E^0 is the Young's modulus for solid material and $\mu_{\min}$ is a minimum fraction of E^0 imposed to ensure that numerical difficulties due to 0 densities are avoided - this is set to a "small" number like $10^{-9} - 10^{-3}$ depending on

the problem. This leads to the following expression for the global stiffness matrix:

$$\mathbf{K}(\boldsymbol{\rho}) = \sum_{e=1}^{n_e} (\mu_{\min} + (1 - \mu_{\min})\, {\rho_e}^p)\, \mathbf{K}_e^0 \tag{2.8}$$

where the summation sign implies finite element summation over all the elements $e = 1, \ldots, n_e$ and $\mathbf{K}_e^0$ is the element stiffness matrix for element e with Young's modulus E^0.

2.3 Solution methods

There are many methods that can be used to solve the minimization problem and update the design variables, and this section covers the two of the gradient-based methods used in the project.

The convergence criteria for the solvers, used throughout the report, is a measure of the maximum change of element densities:

$$\|\boldsymbol{\rho}_{k-1} - \boldsymbol{\rho}_k\|_\infty \leq c_{crit} \tag{2.9}$$

where k denotes the iteration number and c_{crit} is the convergence criteria value, most often set to 10^{-2}, but sometimes changed depending on the problem to be solved.

2.3.1 The Optimality Criterion method

The Optimality Criterion method [OC] is a simple heuristic algorithm to perform the update of the design variables. This method was one of the earlier methods proposed to help solve topology optimization problems, and this reflects in its simplicity. This simplicity, however, also means that it only allows for the imposition of a single constraint. Optimization problem 2.1 can, therefore, be simplified to:

$$\begin{aligned} \min \quad & f_0(\boldsymbol{\rho}) = \mathbf{f}^T\mathbf{u} = \mathbf{u}^T\mathbf{K}(\boldsymbol{\rho})\,\mathbf{u} \\ \text{s.t.} \quad & f_1(\boldsymbol{\rho}) = \frac{\sum_{e=1}^{n_e}(\rho_e V_e)}{V^*} - 1 \leq 0 \\ & 0 \leq \rho_e \leq \rho_{\max}\,,\; e = 1, \ldots, n_e \end{aligned} \tag{2.10}$$

for the minimum compliance optimization problem subject to a volume constraint.

The OC method uses a bisection scheme to find the solution, λ, to a Lagrange function for the single constraint function, and uses this together with the sensitivities of the objective and constraint functions, to update the design variables using the following heuristic update rule:

$$\rho_e = \begin{cases} 0 & \text{if } \rho_{e,\text{old}}(B_e)^\eta \leq 0 \\ \rho_{e,\text{old}}(B_e)^\eta & \text{if } 0 < \rho_{e,\text{old}}(B_e)^\eta < \rho_{\max} \\ \rho_{\max} & \text{if } \rho_{\max} \leq \rho_{e,\text{old}}(B_e)^\eta \end{cases} \tag{2.11}$$

where η is a damping parameter, typically less than 1 (set to 0.5 throughout this report), and B_e is defined as follows:

$$B_e = -\frac{\frac{\partial f_0}{\partial \rho_e}}{\lambda \frac{\partial f_1}{\partial \rho_e}} \tag{2.12}$$

2.3.2 The Method of Moving Asymptotes

The Method of Moving Asymptotes [MMA] is an optimization algorithm that allows for more flexibility than the OC method. The MMA algorithm was published by K. Svanberg in 1987 [13] and is now a widely used algorithm for topology optimization, which is freely available for educational and research purposes by contacting Svanberg[1]. It is based on the concept of convex approximations and, therefore, MMA approximates the actual optimization problem, by generating a subproblem where the objective and constraint functions are replaced by approximating explicit functions, based on the function values and sensitivities found using FEA. It is called the method of "moving asymptotes" because the approximating function is restricted by asymptotes closing in on the area of interests, as the algorithm iterates for a solution.

2.3.3 The Globally Convergent version of MMA

The Globally Convergent version of MMA [GCMMA] was also formulated by K. Svanberg [14]. The global convergence of this method is obtained through inner iterations of MMA, within outer iterations of MMA. The inner iterations are used to obtain conservatism, which is defined as when:

$$\tilde{f}_i^{(k)}(\hat{\mathbf{x}}^{(k)}) \geq f_i(\hat{\mathbf{x}}^{(k)}) \quad \text{for } i = 0, 1, \ldots, m \tag{2.13}$$

where f_i are the original functions, $\tilde{f}_i^{(k)}$ are the approximating functions and $\hat{\mathbf{x}}^{(k)}$ is the solution to the approximating subproblem - the last two at iteration k.

For more information on the MMA and GCMMA algorithm, the already cited references should be looked at, but also Svanberg's notes [15] on MMA and GCMMA are useful. When GCMMA is used to solve an optimization problem, the number of outer iterations and total number of iterations (outer and inner) will be written in the following format: $n_{it,out}$ ($n_{it,tot}$).

[1] http://www.math.kth.se/∼krille/

2.4 Filtering

There are several reasons as to why some form of filtering of the design distribution is beneficial during topology optimization. A few will be covered in this section, and thereafter the relevant types of filtering are described.

Checkerboards

One problem that exists is the appearance of the so-called checkerboards in optimization solutions, where alternating solid and void elements create areas of solid elements connected at the corners only. These occur due to the fact that the optimization procedure always exploits faults and weaknesses in the numerical modelling. The checkerboard pattern results in a structure that is artificially stiff, and the optimization procedure, therefore, takes advantage of this. Another closely related problem is the appearance of single-node pivots as seen in figure 2.1.

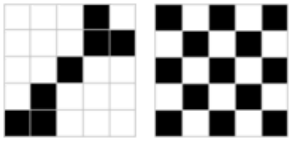


Figure 2.1 – Illustration of checkerboards and single-node pivots.

Mesh-dependence

Another reason for the need of filtering, is to create mesh-independent solutions. Topology optimization usually results in mesh-dependent solutions as a finer mesh allows for the generation of finer truss-like solutions with more holes in them, which generally increases the efficiency of the material use. This is not desirable as an increased resolution and accuracy of the FEM solution then will not go hand-in-hand with an increased resolution and accuracy of the *same* optimal solution.

There are many different ways to fix the complications listed above, most of which are described in Bendsøe and Sigmund's book [24] and the comprehensive investigation of various filter types in Sigmund's paper [12]. The filtering throughout this report is done by looking at the "neighbourhood" of the individual e'th element, which is defined below as the set of elements with centres within the filter radius, R:

$$N_e = \{i \mid \|\mathbf{x}_i - \mathbf{x}_e\| \leq R\} \tag{2.14}$$

where R is the filter radius and $\mathbf{x}_i$ is the spatial location of the element i.

2.4.1 Sensitivity filtering

Filtering of sensitivities is done by updating the element sensitivities with a "weighted average" of the sensitivities of the elements within the predefined neighbourhood (equation 2.14).
The sensitivities are updating using:

$$\frac{\widetilde{\partial f}}{\partial \rho_e} = \frac{\sum_{i \in N_e} w(\mathbf{x}_i) \frac{\rho_i}{v_i} \frac{\partial f}{\partial \rho_i}}{\frac{\rho_e}{v_e} \sum_{i \in N_e} w(\mathbf{x}_i)} \tag{2.15}$$

where v_i is the volume of element i and $w(\mathbf{x}_i)$ is weighting function, which throughout this project is defined as the following linearly decaying, cone-shaped, function:

$$w(\mathbf{x}_i) = R - \|\mathbf{x}_i - \mathbf{x}_e\| \tag{2.16}$$

2.4.2 Density filtering

Filtering of densities is done by updating the element densities with a "weighted average" of the densities of the elements within the predefined neighbourhood (equation 2.14).
The densities are updating using:

$$\tilde{\rho}_e = \frac{\sum_{i \in N_e} w(\mathbf{x}_i) v_i \rho_i}{\sum_{i \in N_e} w(\mathbf{x}_i) v_i} \tag{2.17}$$

where $w(\mathbf{x}_i)$ is the weighting function defined in equation 2.16.

The filtered densities, $\tilde{\boldsymbol{\rho}}$, are now the physically meaningful variables, whereas $\boldsymbol{\rho}$ is now just a non-physical design variable. As it is $\boldsymbol{\rho}$ that is updated using the optimization algorithms, the sensitivities have to be updated using the chain rule:

$$\frac{\partial f}{\partial \rho_e} = \sum_{i \in N_e} \frac{\partial f}{\partial \tilde{\rho}_i} \frac{\partial \tilde{\rho}_i}{\partial \rho_e} \tag{2.18}$$

where the sensitivity of the filtered density with respect to the non-physical design variable is:

$$\frac{\partial \tilde{\rho}_i}{\partial \rho_e} = \frac{w(\mathbf{x}_e) v_e}{\sum_{j \in N_i} w(\mathbf{x}_j) v_j} \tag{2.19}$$

2.5 Optimization with respect to steady-state heat transfer

Topology optimization wrt. steady-state heat transfer is very similar to that of static mechanical problems. However, due to temperature being a scalar quantity, the FE problem for heat transfer is simpler and not as computationally demanding per node, due to there being only one degree of freedom [DOF] per node. Therefore, for the same discretised domain, the thermal problem will be faster to solve and take up less storage space than the equivalent static mechanical problem.

The objective function commonly used for purely thermal topology optimization problems is the thermal compliance. This is analogous to the mechanical compliance, in that it is the inverse of the thermal stiffness:

$$f_0(\boldsymbol{\rho}) = \mathbf{f_t}^T \mathbf{t} \tag{2.20}$$
$$= \mathbf{t}^T \mathbf{K_t}(\boldsymbol{\rho})\, \mathbf{t} \tag{2.21}$$

where $\mathbf{f_t}$ and $\mathbf{t}$ are the thermal load and temperature vectors, respectively, from the finite element equilibrium equation for steady-state heat transfer:

$$\mathbf{K_t}(\boldsymbol{\rho})\, \mathbf{t} = \mathbf{f_t} \tag{2.22}$$

where $\mathbf{K_t}$ is the thermal stiffness matrix as a function of the density variable distribution, $\boldsymbol{\rho}$, as defined below.

The thermal expansion coefficient is assumed to be independent of the density and the thermal conductivity is parametrised to be dependent of the variable density using the following equation:

$$k(\rho_e) = (\mu_{\min} + (1 - \mu_{\min})\, {\rho_e}^p)\, k^0 \tag{2.23}$$

This leads to the following expression for the global thermal stiffness matrix:

$$\mathbf{K}_t(\boldsymbol{\rho}) = \sum_{e=1}^{n_e} (\mu_{\min} + (1 - \mu_{\min})\, {\rho_e}^p)\, \mathbf{K}^0_{t,e} \tag{2.24}$$

where the summation sign implies finite element summation over all the elements $e = 1, \ldots, n_e$ and $\mathbf{K}^0_{t,e}$ is the element stiffness matrix for element e with thermal conductivity k^0.

Chapter 3

Coupling of static mechanics and steady-state heat transfer

The fields are assumed weakly coupled; that is the thermal field will give rise to thermal strains, but the displacement field doesn't give rise to changes in temperature. Therefore, the coupling of static mechanics and steady-state heat transfer is relatively simple when it comes to the FEM implementation. Firstly, the temperature field is solved for the given thermal boundary conditions and loads. The thermal strain is then found, converted to a load vector and added to any existing loads. Finally, the displacement field is solved for the given boundary conditions and loads.

The well-known equilibrium equations for thermal and mechanical analysis are:

$$\mathbf{K}_t\mathbf{t} = \mathbf{f}_t \tag{3.1}$$

$$\mathbf{K}\mathbf{u} = \mathbf{f}(\mathbf{t}) \tag{3.2}$$

The mechanical load vector, $\mathbf{f}$, depends on the temperature field, $\mathbf{t}$, in that it is defined as the addition of purely mechanical loads and the thermal-strain load:

$$\mathbf{f}(\mathbf{t}) = \mathbf{f}_{\text{mech}} + \mathbf{f}_{\text{therm}}(\mathbf{t}) \tag{3.3}$$

The thermal load vector for an individual element is defined as:

$$\mathbf{f}^e_{\text{therm}}(\mathbf{t}) = \int_{V_e} \mathbf{B}^T\mathbf{C}\boldsymbol{\varepsilon_0} dV_e \tag{3.4}$$

where $\mathbf{B}$ is the strain-displacement matrix and $\mathbf{C}$ is the constitutive matrix for the mechanical problem. The thermally-induced strain is:

$$\boldsymbol{\varepsilon_0} = \boldsymbol{\alpha} T_e = \left\{ \begin{array}{c} \alpha \\ \alpha \\ 0 \end{array} \right\} T_e \tag{3.5}$$

where α is the coefficient of thermal expansion and T_e is the temperature change of the element. The element temperature is interpolated using the thermal shape function matrix and the nodal temperatures for the element:

$$T_e = \mathbf{N}_t \mathbf{t}_e \tag{3.6}$$

This leads to the thermal-strain load being expressed as:

$$\mathbf{f}^e_{\text{therm}}(\mathbf{t}) = \int_{V_e} \mathbf{B}^T \mathbf{C} \boldsymbol{\alpha} \mathbf{N}_t \mathbf{t}_e dV_e \tag{3.7}$$

which can be further simplified to:

$$\mathbf{f}^e_{\text{therm}}(\mathbf{t}) = \mathbf{A}_e \mathbf{t}_e \tag{3.8}$$

when the coupling matrix, $\mathbf{A}_e$, is defined as:

$$\mathbf{A}_e = \int_{V_e} \mathbf{B}^T \mathbf{C} \boldsymbol{\alpha} \mathbf{N}_t dV_e \tag{3.9}$$

The global thermally-induced load can thereby be expressed as the FE-summation over all the element contributions:

$$\mathbf{f}_{\text{therm}}(\mathbf{t}) = \sum_{e=1}^{N} \mathbf{A}_e \mathbf{t}_e = \mathbf{A}\mathbf{t} \tag{3.10}$$

where $\mathbf{A}_e$, $\mathbf{A}$, $\mathbf{t}_e$ and $\mathbf{t}$ are the local and global coupling matrices and temperature vectors respectively. The coupling matrix depends on the design variables as described in section 2.2.

Although the coupling of the fields is relatively easy for the FEA, the sensitivity analysis, of the objective and constraint functions, often become more complicated as the functions now may depend on both fields. The adjoint method for sensitivity analysis is often used as this helps avoid having to calculate the direct derivatives of the physical fields.

3.1 Adjoint sensitivity analysis

The needed sensitivities, for the various coupled problems throughout this report, are all found using the adjoint method as described below (and in [23], [24]). The sensitivity derived below can be used for any objective function that can be written in the form of a vector multiplied by the displacement field, but has been carried out for use with the maximisation of a single nodal displacement, used in section 7.1. To obtain a single nodal displacement, the vector $\mathbf{l}$ is specified to have 0's in all entries except for a 1 at the nodal DOF to be maximized. The other coupled sensitivity analysis used in this project can be found in appendix B.

$$\begin{aligned} \min \quad & \Phi = -\mathbf{l}^T\mathbf{u} \\ \text{s.t.} \quad & \mathbf{K}_t(\boldsymbol{\rho})\,\mathbf{t} = \mathbf{f}_t \\ & \mathbf{K}(\boldsymbol{\rho})\,\mathbf{u} = \mathbf{f}(\boldsymbol{\rho},\mathbf{t}) \\ & 0 \le \rho_e \le \rho_{\max}\,,\; e = 1,\dots,N \end{aligned} \tag{3.11}$$

Two null terms are subtracted from the objective function:

$$\Phi = -\mathbf{l}^T\mathbf{u} - \boldsymbol{\lambda}_1{}^T(\mathbf{K}_t\mathbf{t} - \mathbf{f}_t) - \boldsymbol{\lambda}_2{}^T(\mathbf{K}\mathbf{u} - \mathbf{f}) \tag{3.12}$$

where $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ are fixed adjoint vectors that can be chosen freely if the equilibrium equations are satisfied.
The sensitivity of the objective function, is found by differentiating the objective function wrt. a design change:

$$\begin{aligned} \frac{\partial\Phi}{\partial\rho_e} \;=\; & \left(-\mathbf{l}^T - \boldsymbol{\lambda}_2{}^T\mathbf{K}\right)\frac{\partial\mathbf{u}}{\partial\rho_e} + \left(-\boldsymbol{\lambda}_1{}^T\mathbf{K}_t + \boldsymbol{\lambda}_2{}^T\frac{\partial\mathbf{f}}{\partial\mathbf{t}}\right)\frac{\partial\mathbf{t}}{\partial\rho_e} \\ & + \boldsymbol{\lambda}_1{}^T\left(\frac{\partial\mathbf{f}_t}{\partial\rho_e} - \frac{\partial\mathbf{K}_t}{\partial\rho_e}\mathbf{t}\right) + \boldsymbol{\lambda}_2{}^T\left(\frac{\partial\mathbf{f}}{\partial\rho_e} - \frac{\partial\mathbf{K}}{\partial\rho_e}\mathbf{u}\right) \end{aligned} \tag{3.13}$$

By choosing $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ as the solutions to the following adjoint problems, it can be avoided to find the direct derivatives of the field variables.

$$\mathbf{K}\boldsymbol{\lambda}_2 = -\mathbf{l} \tag{3.14}$$

$$\mathbf{K}_t\boldsymbol{\lambda}_1 = \frac{\partial\mathbf{f}}{\partial\mathbf{t}}^T\boldsymbol{\lambda}_2 \tag{3.15}$$

With $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ as solutions to the adjoint problems above, the expression for the sensitivity can be simplified to:

$$(3.13) \Longrightarrow \frac{\partial\Phi}{\partial\rho_e} = \boldsymbol{\lambda}_1{}^T\left(\frac{\partial\mathbf{f}_t}{\partial\rho_e} - \frac{\partial\mathbf{K}_t}{\partial\rho_e}\mathbf{t}\right) + \boldsymbol{\lambda}_2{}^T\left(\frac{\partial\mathbf{f}}{\partial\rho_e} - \frac{\partial\mathbf{K}}{\partial\rho_e}\mathbf{u}\right) \tag{3.16}$$

where the sensitivities of the stiffness matrices and load vectors are easily found from the implemented finite element formulation (equations 2.8, 2.24, 3.10, 4.18 and 4.23).

Chapter 4

Convection

This chapter contains general theory of convection and details on the FEM implementation. Also the formulation of the design-dependent convection is made and an analytical study is carried out.

4.1 General convection

Convection is the transfer of energy from a solid object to a surrounding fluid. Most often this is caused by the fluid moving along the object, called forced convection, but can also occur to a surrounding stagnant fluid, called free convection. Forced convection is frequently found in practical problems and is a complex problem within fluid dynamics. The characteristics of the flow determines how much energy is transferred from the object to the moving fluid. Therefore, parameters such as speed and properties of the moving fluid, and size, shape and properties of the solid object, play a huge role in this phenomenon. As these often vary across the surface of the object, the problem quickly becomes very complex. However, this process can be simplified significantly by defining an average convection heat transfer coefficient across the surface of the object. Which leads to the following expression for the energy transferred by convection:

$$q_{conv} = hA_s(T_s - T_\infty) \tag{4.1}$$

where h is the convection heat transfer coefficient, A_s is the surface area, T_s is the surface temperature and T_∞ is the fluid temperature. Throughout this project this simplified model for forced convection will be adopted as it is suited for pure heat transfer analysis. A more thorough description of the workings of convection is, therefore, outside the scope of this project. The convection heat transfer coefficient can be estimated relatively easily, or found in tables and textbooks, for various flows and object shapes.

There are two different kinds of convection implemented, that differ in the area affected and the number of nodes involved.

Top and bottom convection [TBC] is when the two lateral surfaces of the plane problem is subjected to convection. Both the top and bottom areas of the elements are subjected to convection and the load affects all nodes of the elements.

Side convection [SC] is when the surfaces perpendicular to the plane is subjected to convection. Only a single side of the element, per load, are subjected to convection and the load, therefore, only affects the two nodes on the corresponding side.

4.2 FEM implementation

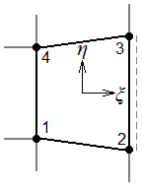


Figure 4.1 – Boundary

For an element subject to convection on the side of the 2nd and 3rd node, as shown in figure 4.1, the isoparametric formulation for the boundary convection matrix is as follows:

$$\mathbf{K}_h^e = \int_S \mathbf{N}_t{}^T \mathbf{N}_t \, h \, dS = \int_{-1}^{1} \mathbf{N}_t{}^T \mathbf{N}_t \, h \, t \, J d\eta = \frac{h t L_{23}}{6} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.2}$$

where $\mathbf{N}_t$ is the shape function matrix for the temperature field, h is the convection heat transfer coefficient, t and J is the thickness and the Jacobian of the element respectively, and L_{23} is the length of the affected side. The corresponding thermal load vector is formulated as follows:

$$\mathbf{f}_h^e = \int_S \mathbf{N}_t{}^T \, h T_\infty \, dS = \int_{-1}^{1} \mathbf{N}_t{}^T \, h T_\infty \, t J d\eta = T_\infty \frac{h t L_{23}}{2} \begin{Bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{Bmatrix} \tag{4.3}$$

where T_∞ is the temperature of the surrounding fluid.

The isoparametric formulation for the convection matrix for a single lateral surface of the plane problem is as follows:

$$\mathbf{K}_{h,ls}^e = \int_S \mathbf{N}_t{}^T \mathbf{N}_t \, h \, dS = \int_{-1}^{1} \int_{-1}^{1} \mathbf{N}_t{}^T \mathbf{N}_t \, h \, J d\eta \, d\xi \tag{4.4}$$

This is not quite as easy to reduce to a simple matrix that can be shown, as it depends on the size and shape of the element, and not only the length of a single edge. But there will be values present in all of the diagonal entries, and also some off-diagonal, as all nodes are affected by convection. The corresponding thermal load vector is formulated as:

$$\mathbf{f}_h^e = \int_S \mathbf{N}_t{}^T \, h T_\infty \, dS = \int_{-1}^{1} \int_{-1}^{1} \mathbf{N}_t{}^T \, h T_\infty \, d\eta \, d\xi \tag{4.5}$$

Lumped matrices

Lumping (diagonalisation) of matrices is a practise much used within numerical dynamics, where the mass matrix is lumped to save on computation time and difficulty. Lumping of the convection matrix is here used to avoid numerical oscillations in the temperature response around the convection temperature. This is suggested by Bruns [21] and in his paper it is shown to remove these oscillations. However, this is still looked upon further in section 4.4, as part of the analytical study. The off-diagonal terms in the consistent convection matrix appear due to the numerical integration taking place in the integration points of the element, and not at the nodes. The diagonalised, or lumped, matrix is obtained by collecting all of the contribution in the diagonal of the matrix, that is on the nodes of the element. The lumping scheme used in this project is as follows:

$$\tilde{\mathbf{M}}_{i,i} = \sum_{j=1}^{N} \mathbf{M}_{i,j} \tag{4.6a}$$

$$\tilde{\mathbf{M}}_{i,j} = 0 \qquad \text{for } i \neq j \tag{4.6b}$$

where $\mathbf{M}$ is a N by N matrix, i and j denote the column and row number respectively and $\tilde{\mathbf{M}}$ is the lumped matrix.

As an example, lumping changes the boundary convection matrix, equation 4.2, for a side convection load to:

$$\mathbf{K}_h^e = \frac{htL_{23}}{2} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.7}$$

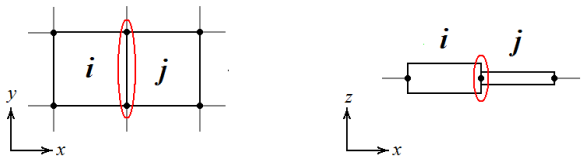
4.3 Design-dependent convection

Normally, it is only possible to impose convection on the boundaries of the design-domain. That is on the plane surfaces (TBC) and on the outer most edges around the design-domain (SC). Of course, it is also possible to impose side convection on the edges of the internal elements, but this would be constant and, therefore, design-independent convection. The convection formulation, however, ought to be design-dependent in that the side convection only should be applied on the boundaries between the design and the surrounding fluid (represented by 0-density elements). The complication arises because when using topology optimization, the position and size of inner holes and boundaries are not known beforehand and change during the iterative progress for a solution.

In research, there have been several suggestions to how this can be obtained. Ananthasuresh and Yin [22] suggests a density-based peak interpolation-function, Yoon and Kim [18] suggests a special type of discontinuous elements and Iga et.al [20] suggests a density-based smeared-out Hat function and an interpolation-function where several coefficients are based on numerical fluid simulation models. Lastly, Bruns [21] suggests, but does not test, a density-dependent interpolation-function that seems like the most viable candidate for the problem at hand. This method is further discussed and outlined below.

4.3.1 Internal Side Convection

Internal side convection [ISC] has been chosen to describe the side convection taking place within the boundaries of the discretised domain.



(a) Plane view

(b) 'Variable-thickness representation'

Figure 4.2 – The interface between two elements affected by ISC seen from two angles.

Bruns [21] suggests a design-dependent interpolation function for the convection heat transfer coefficient, where half of the contribution is added to each element at the interface. For the two elements shown in figure 4.2, the interpolated convection heat transfer coefficient is found using the following:

$$h_i = \frac{1}{2} g(\rho_i - \rho_j)\, \bar{h} \tag{4.8a}$$

$$h_j = \frac{1}{2} g(\rho_j - \rho_i)\, \bar{h} \tag{4.8b}$$

where the function $g(x) \approx |x|$. This interpolation function is adopted in this project, and as the absolute-function is non-differentiable at $x = 0$, it is approximated by the following function:

$$g(x) = \sqrt{x^2 + \epsilon^2} - \epsilon \approx |x| \tag{4.9}$$

As ϵ goes towards 0, the function value will go towards the absolute value of x, so a small value is used for ϵ (10^{-5} throughout this project). This way of interpolating the convection heat transfer is very intuitive, when the problem is seen as a variable-thickness problem, as laid out in figure 4.2b. The difference in the density, or thickness, gives the remaining free length of the edge exposed to convection.

The following FE formulation of the convection matrix and load is implemented:

$$\mathbf{K}^e_{h,ISC} = \frac{1}{2} \sum_{j=1}^{n_{int}} g(\rho_e - \rho_j)\, \mathbf{K}^e_{j,\bar{h}} \tag{4.10a}$$

$$\mathbf{f}^e_{h,ISC} = \frac{1}{2} \sum_{j=1}^{n_{int}} g(\rho_e - \rho_j)\, \mathbf{f}^e_{j,\bar{h}} \tag{4.10b}$$

where ρ_j is the density of the j'th interface element (not the j'th global element), the subscript "$j, \bar{h}$" indicates the matrix and vector evaluated for the nominal convection heat transfer coefficient $\bar{h}$ at the corresponding side of the j'th interface, and n_{int} is the number of interfaces for element e. The number of interfaces for each element is found at the start of the optimization, where all elements are run through and the corresponding edges and interface elements are saved for all future FE-function calls.

The sensitivities of the ISC convection matrix to a design change are a little more tricky than usual, as the element contributions not only depend on the element density, but also the densities of the adjacent interface elements. Firstly, $\tau_e = \rho_e - \rho_j$ is substituted and a differentiation wrt. this variable is performed:

$$\begin{aligned} \mathbf{K}^e_{h,ISC} &= \frac{1}{2} \sum_{j=1}^{n_{int}} g(\tau_e)\, \mathbf{K}^e_{j,\bar{h}} \\ \Updownarrow \frac{\partial}{\partial \tau_e} \mathbf{K}^e_{h,ISC} &= \frac{1}{2} \sum_{j=1}^{n_{int}} \left. \frac{\partial g}{\partial \tau_e} \right|_{x=\tau_e} \mathbf{K}^e_{j,\bar{h}} \end{aligned} \tag{4.11}$$

Since:

$$\mathbf{K}_{h,ISC} = \sum_{e=1}^{n_e} \mathbf{K}^e_{h,ISC} \tag{4.12}$$

a change in the e'th element will lead to a change in both the e'th elements convection matrix, but also the convection matrices of the corresponding interface elements. That is, the change will be double that of the single contribution to the element convection matrix, and therefore:

$$\frac{\partial}{\partial \rho_e}\mathbf{K}_{h,ISC} = \sum_{j=1}^{n_{int}} g'(\rho_e - \rho_j)\, \mathbf{K}^e_{j,\bar{h}} \tag{4.13}$$

where $g'(x)$ is the function defined in equation 4.9 differentiated wrt. to the input variable x:

$$g'(x) = \frac{\partial g}{\partial x} = \frac{x}{\sqrt{x^2 + \epsilon^2}} \tag{4.14}$$

The sensitivities of the respective thermal load is found using the same method to:

$$\frac{\partial}{\partial \rho_e}\mathbf{f}_{h,ISC} = \sum_{j=1}^{n_{int}} g'(\rho_e - \rho_j)\, \mathbf{f}^e_{j,\bar{h}} \tag{4.15}$$

The analytical study in section 4.4, indicates that it can be beneficial to introduce a minimum convection fraction analogous to the minimum stiffness introduced in the SIMP model. The interpolated convection heat transfer coefficients then become:

$$h_i = \frac{1}{2}(h_{min} + (1 - h_{min})\, g(\rho_i - \rho_j))\, \bar{h} \tag{4.16a}$$

$$h_j = \frac{1}{2}(h_{min} + (1 - h_{min})\, g(\rho_j - \rho_i))\, \bar{h} \tag{4.16b}$$

which in turn leads to equation 4.10 changing to:

$$\mathbf{K}^e_{h,ISC} = \frac{1}{2}\sum_{j=1}^{n_{int}}(h_{min} + (1 - h_{min})\, g(\rho_e - \rho_j))\, \mathbf{K}^e_{j,\bar{h}} \tag{4.17a}$$

$$\mathbf{f}^e_{h,ISC} = \frac{1}{2}\sum_{j=1}^{n_{int}}(h_{min} + (1 - h_{min})\, g(\rho_e - \rho_j))\, \mathbf{f}^e_{j,\bar{h}} \tag{4.17b}$$

and equations 4.13 and 4.15 to:

$$\frac{\partial}{\partial \rho_e}\mathbf{K}_{h,ISC} = \sum_{j=1}^{n_{int}}(1 - h_{min})\, g'(\rho_e - \rho_j)\, \mathbf{K}^e_{j,\bar{h}} \tag{4.18a}$$

$$\frac{\partial}{\partial \rho_e}\mathbf{f}_{h,ISC} = \sum_{j=1}^{n_{int}}(1 - h_{min})\, g'(\rho_e - \rho_j)\, \mathbf{f}^e_{j,\bar{h}} \tag{4.18b}$$

The effect of this minimum applied convection on the temperature distribution will be explained in section 4.4 and the effect on the resulting topologies after optimization will be investigated in chapter 8.

4.3.2 Boundary Side Convection

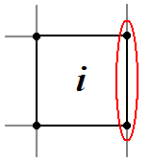


Figure 4.3 – The boundary of an element subject to convection, seen in the plane view.

Boundary side convection [BSC] is the convection taking place at the boundaries of the discretised domain. The active convection heat transfer coefficient is here made dependent of only the density of the single element:

$$h_i = \rho_i \bar{h} \tag{4.19}$$

As with ISC, the possibility of a minimum convection coefficient is introduced, giving:

$$h_i = (h_{min} + (1 - h_{min})\,\rho_i)\,\bar{h} \tag{4.20}$$

The FEM implementation, therefore, becomes:

$$\mathbf{K}^e_{h,BSC} = (h_{min} + (1 - h_{min})\,\rho_e)\,\mathbf{K}^e_{\bar{h}} \tag{4.21}$$

$$\mathbf{f}^e_{h,BSC} = (h_{min} + (1 - h_{min})\,\rho_e)\,\mathbf{f}^e_{\bar{h}} \tag{4.22}$$

and the sensitivities simply become:

$$\frac{\partial}{\partial \rho_e}\mathbf{K}_{h,BSC} = (1 - h_{min})\,\mathbf{K}^e_{\bar{h}} \tag{4.23}$$

$$\frac{\partial}{\partial \rho_e}\mathbf{f}_{h,BSC} = (1 - h_{min})\,\mathbf{f}^e_{\bar{h}} \tag{4.24}$$

4.3.3 Top and Bottom Convection

Top and bottom convection is at first left as independent of the design, that is the convection matrix and load is constant and the sensitivities are, therefore, 0 just as if the load had been e.g. a constant heat flux. As a consequence of the analytic study in section 4.4, it is suggested that it could be possible to make TBC design-dependent by introducing the following function for the convection heat transfer coefficient:

$$h_i = q(\rho_i)\,\bar{h} \tag{4.25}$$

where *q(x)* is given by the following expression:

$$q(x) = \frac{x^2}{x^2 + \epsilon} \tag{4.26}$$

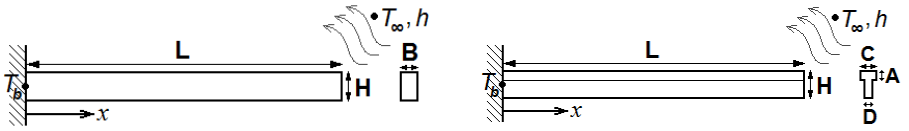
which approximates a discontinuous function with the function value 1 at all x's except for $x = 0$, when ϵ goes towards 0. Another possibility is to use:

$$q(x) = \frac{x^2 + h_{min}\epsilon}{x^2 + \epsilon} \tag{4.27}$$

which would allow for the introduction of a small minimum convection contribution for $x = 0$. Unfortunately, time has not allowed for this to be implemented and tested properly.

4.4 Analytical study

An analytical study has been carried out of the problems depicted in figure 4.4. Analytical solutions have been derived and various simulations have been carried out using the implemented convection FE formulation, to confirm the validity of the implementation.



(a) Heat beam 1 **(b)** Heat beam 2

Figure 4.4 – Problem layout for the analytical study.

Three subproblems are set up, having the convection boundary conditions as laid out in table 4.1. The derivations of the analytical 1D temperature distributions can be found in appendix C.

	Boundary conditions	**Temperature dist.**
a	Top and bottom	Equation 4.28
b	Side	Equation 4.29
c	Top, bottom and side	Equation 4.29

Table 4.1 – Definitions of the subproblems.

$$\theta(x) = \frac{\theta_b}{1 + e^{2mL}}\left(e^{mx} + e^{m(2L-x)}\right) \tag{4.28}$$

$$\theta(x) = \frac{\theta_b}{1 + \frac{km-h}{km+h}e^{-2mL}}\left(\frac{km-h}{km+h}e^{m(x-2L)} + e^{-mx}\right) \tag{4.29}$$

where $\theta(x) = T(x) - T_\infty$, $\theta_b = T_b - T_\infty$, k is the thermal conductivity, h is the convection heat transfer coefficient and m is a problem-dependent parameter defined in equation 4.30.

$$m^2 = \frac{h}{kA_c}\frac{dA_s}{dx} \tag{4.30}$$

where A_c is the cross-sectional area of the beam and $\frac{dA_s}{dx}$ is the area exposed to convection per unit length. The expressions for this parameter, for the different problems, can also be found in appendix C. The effect of adding 0-density elements around the beam, as shown in figure 4.5, on the temperature distribution in the beam and outside the beam, will be investigated.

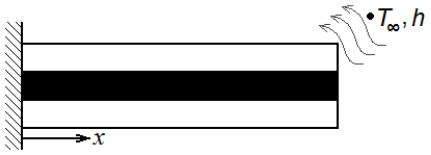


Figure 4.5 – The 'optimization ' layout for the problem.

Heatbeam 1

The dimensions of this problem, as specified in figure 4.4a, are set to $B = 10^{-3}$m, $H = 10^{-2}$m and $L = 0.5$m. The prescribed base temperature is $T_b = 10°$C, the fluid temperature is $T_\infty = 1°$C, the convection heat transfer coefficient is $h = 0.25\,\mathrm{Wm^{-2}K^{-1}}$ and the thermal conductivity of the beam is $k = 143.0\,\mathrm{Wm^{-1}K^{-1}}$.

Figures 4.6 and 4.10 show the results for the beam problem 1a; that is a beam with a rectangular cross-section exposed to only TBC. It is clear from the figures, that the lumping of the convection matrix fixes the numerical oscillations that occur, around the fluid temperature, outside of the beam. It can also be observed that the temperature levels inside the beam are lower, than the analytical and standard FEM solutions. This is likely due to the fact that energy is taken out of the system from the 0-density elements, even though they 'physically' have no surface area that should contribute to the TBC.

It is clear that the temperature distribution is far from correct, when looking at the results for beam problem 1b; the rectangular cross-section beam subject to only SC. More precisely, figures 4.7 and 4.11 show that the temperature distribution in the x-direction is correct and very close to the analytical solution, but the temperature ought to change significantly at the transition from the 1- to 0-density areas. The temperature response in the 0-density area should be close to fluid temperature, but it can be seen that it is more or less the same as in the 1-density area. It is suggested that this can be solved by imposing a small minimum convection contribution on all side convection loads, as laid out in section 4.3.1. As can be seen in figure 4.7d, this seems to solve the problem, however, with a small trade-off in the accuracy of the solution within the beam. The oscillations due to the consistent convection matrices are not quite as clear as for the pure TBC problem, but are still shown to be fixed by lumping in figure 4.7e.

As can be seen in figure 4.14, a value of $h_{\min} = 10^{-3}$ proved to be the value, which gave the best temperature response outside the beam, with the least loss of accuracy of the solution inside the beam. This value has been used for all subsequent optimizations based on this, but a more thorough investigation into the affect of the parameter would be beneficial, to determine whether this is problem dependent.

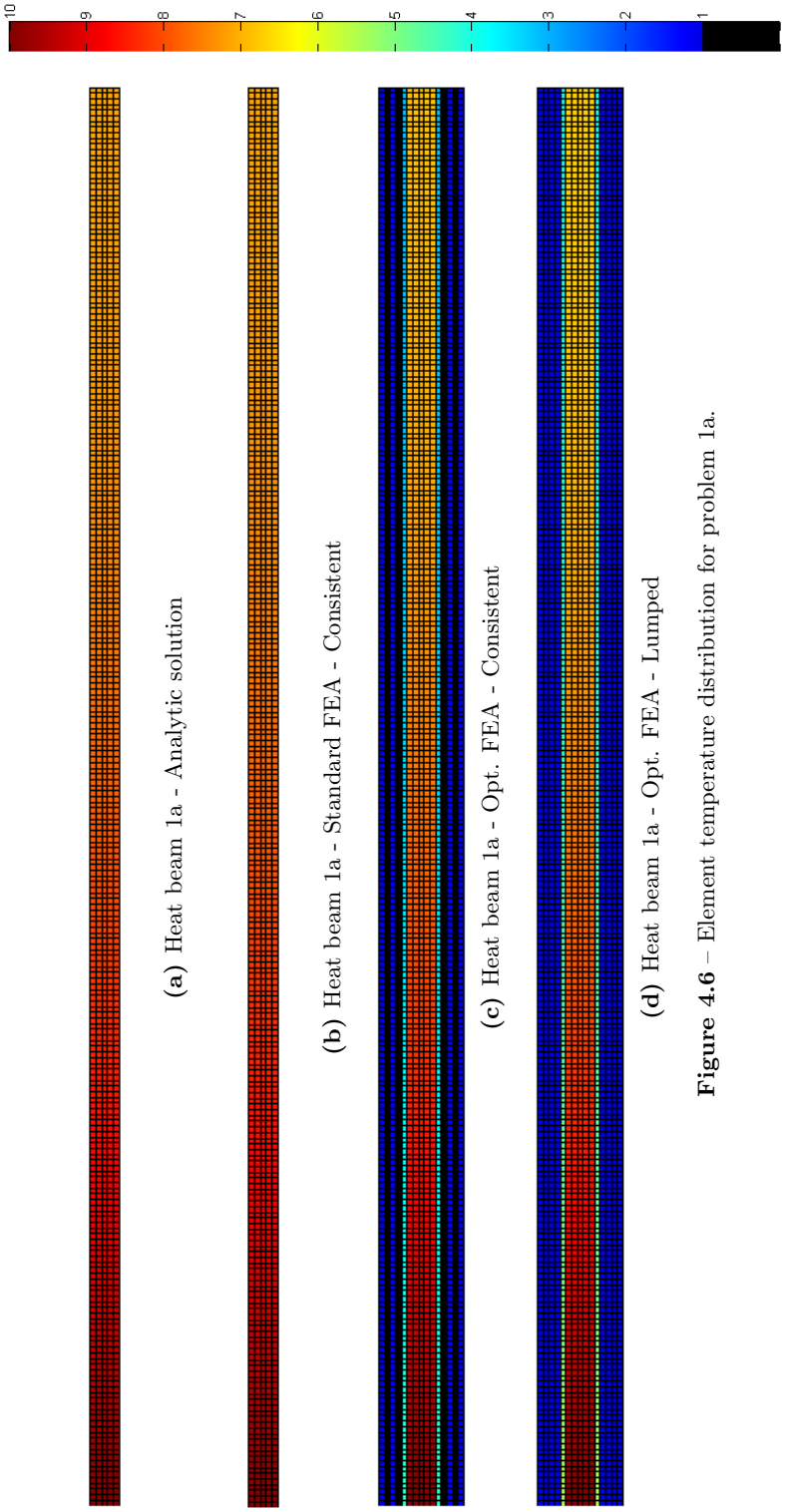
When combining TBC and SC, as in beam problem 1c, it appears that

the TBC drowns the difficulties experienced for the pure SC problem. That is the temperature level outside of the beam is close to the fluid temperature. The important thing to notice in this problem, is that the minimum convection coefficient imposed for the SC, does not appear to change the result of the temperature distribution much - as can be seen in figures 4.12d and 4.12e. These two things are most likely due to the fact that the magnitude of the TBC load is much larger, as the area involved in this load is much larger than for the SC load.

Heatbeam 2

The dimensions of this problem, as specified in figure 4.4b, are set to $C = 10^{-3}$m, $D = 0.5 \times 10^{-3}$m, $H = 10^{-2}$m, $A = 0.4 \times 10^{-2}$m, and length $L = 0.5$m. The rest of the parameters are the same as for heatbeam 1.

The second beam problem has been run for all the subproblems, but the most important effect is observed in subproblem a and therefore only this is shown. Problem 2a is the T-beam subjected to only TBC and the results are shown in figures 4.9 and 4.13. The problem has been run for both the 'optimization layout' as described earlier, but also what has been chosen to be called a 'variable-thickness layout'. Here only the beam is modelled, but instead of running a standard FEA with two sets of elements with different thicknesses, an analysis is run where the density of the lower part of the T-beam has been set to 0.5 . The same effect as with problem 1a can be observed, that is when the beam is surrounded by 0-density elements, the temperature level inside the beam is much lower than the analytical solution. The results for the 'variable-thickness' analyses show that these are very close to the analytical solution, and therefore it can be concluded that the problem is the addition of 0-density elements around the beam. It is, therefore, suggested that a design-dependent TBC be implemented, as touched upon in section 4.3.3, to address this problem. Here TBC load on 0-density elements will be removed or set to a minimum, as with ISC, and less energy would be removed from the system as a result. Initial tests show that this could indeed fix the accuracy of the temperature distribution in the beam, but unfortunately time constraints have not allowed for this to be fully implemented and tested.



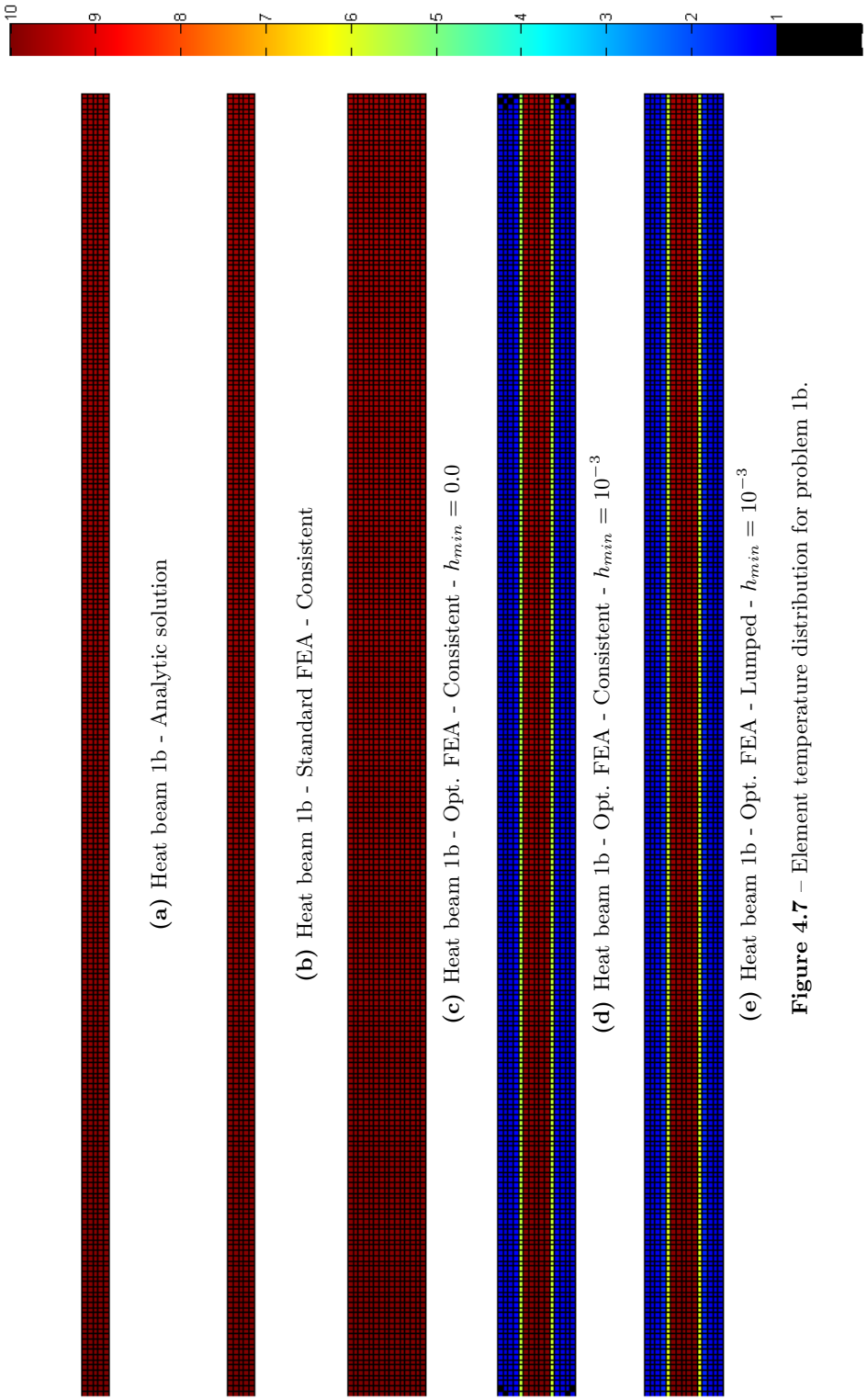
(a) Heat beam 1a - Analytic solution

(b) Heat beam 1a - Standard FEA - Consistent

(c) Heat beam 1a - Opt. FEA - Consistent

(d) Heat beam 1a - Opt. FEA - Lumped

Figure 4.6 – Element temperature distribution for problem 1a.



(a) Heat beam 1b - Analytic solution

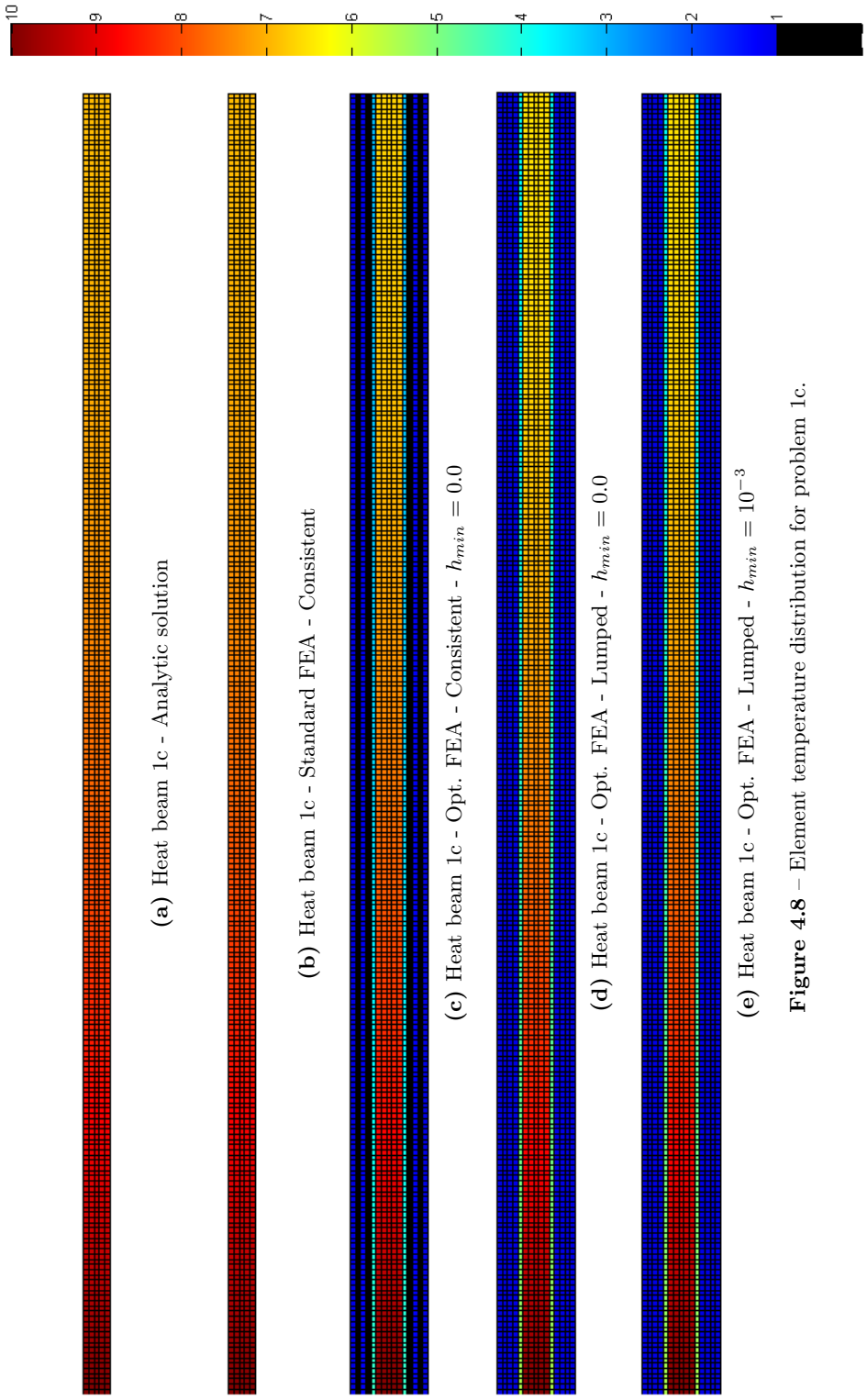
(b) Heat beam 1b - Standard FEA - Consistent

(c) Heat beam 1b - Opt. FEA - Consistent - $h_{min} = 0.0$

(d) Heat beam 1b - Opt. FEA - Consistent - $h_{min} = 10^{-3}$

(e) Heat beam 1b - Opt. FEA - Lumped - $h_{min} = 10^{-3}$

Figure 4.7 – Element temperature distribution for problem 1b.



(a) Heat beam 1c - Analytic solution

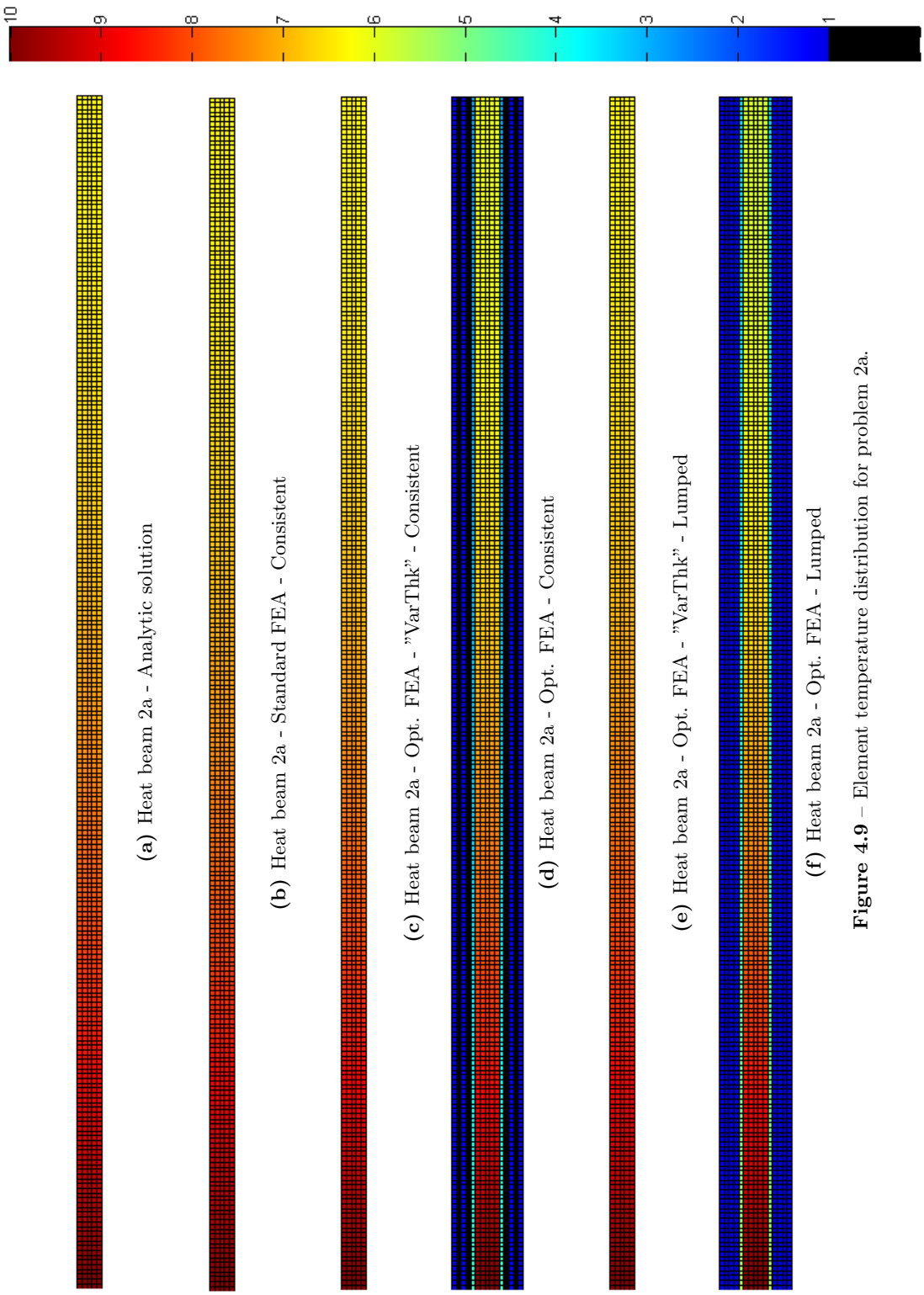
(b) Heat beam 1c - Standard FEA - Consistent

(c) Heat beam 1c - Opt. FEA - Consistent - $h_{min} = 0.0$

(d) Heat beam 1c - Opt. FEA - Lumped - $h_{min} = 0.0$

(e) Heat beam 1c - Opt. FEA - Lumped - $h_{min} = 10^{-3}$

Figure 4.8 – Element temperature distribution for problem 1c.



(a) Heat beam 2a - Analytic solution

(b) Heat beam 2a - Standard FEA - Consistent

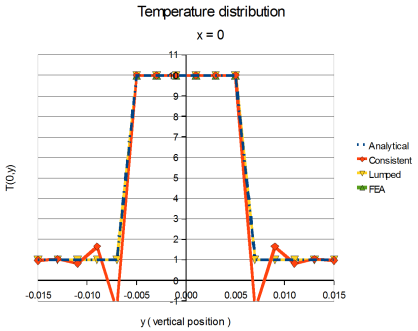
(c) Heat beam 2a - Opt. FEA - "VarThk" - Consistent

(d) Heat beam 2a - Opt. FEA - Consistent

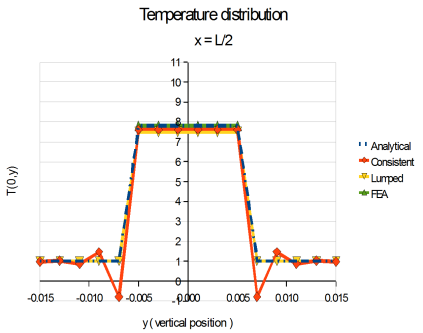
(e) Heat beam 2a - Opt. FEA - "VarThk" - Lumped

(f) Heat beam 2a - Opt. FEA - Lumped

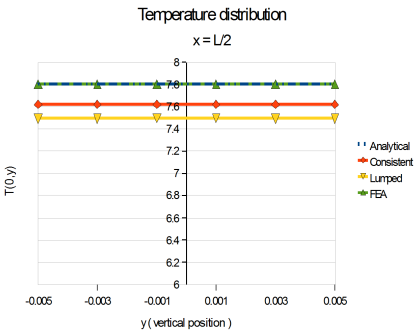
Figure 4.9 – Element temperature distribution for problem 2a.



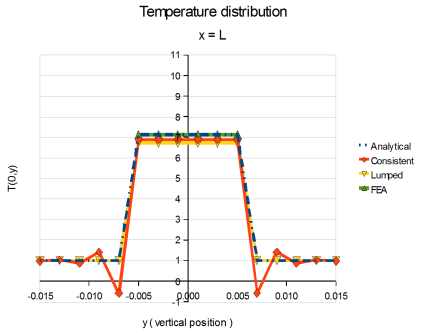
(a) Heat beam 1a - $x = 0$



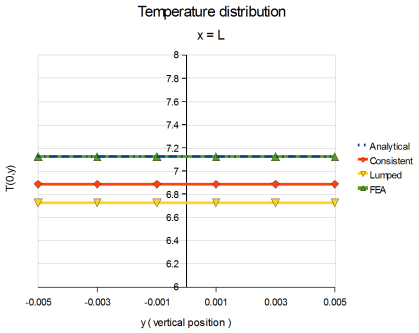
(b) Heat beam 1a - $x = \frac{L}{2}$



(c) Heat beam 1a - $x = \frac{L}{2}$

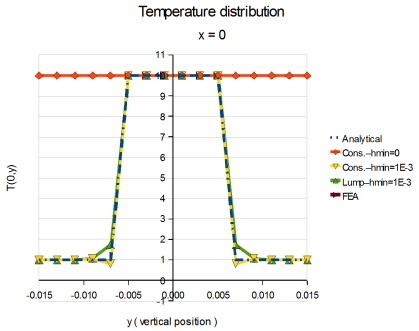


(d) Heat beam 1a - $x = L$

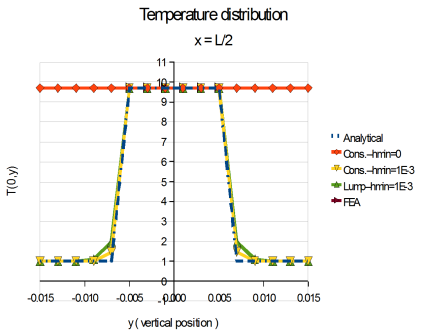


(e) Heat beam 1a - $x = L$

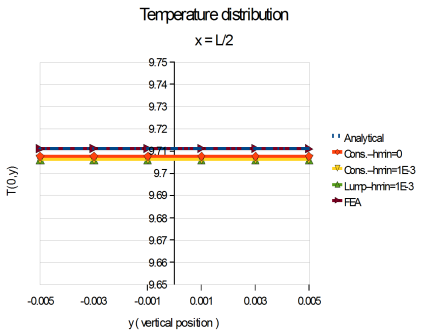
Figure 4.10 – Nodal temperature distribution for problem 1a.



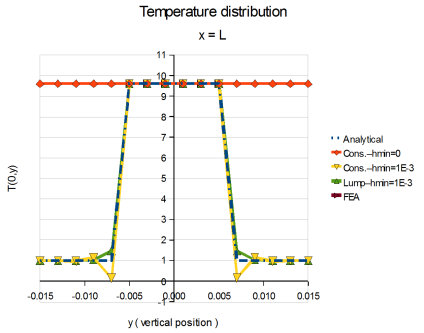
(a) Heat beam 1b - $x = 0$



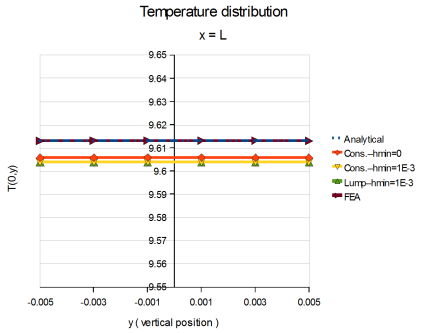
(b) Heat beam 1b - $x = \frac{L}{2}$



(c) Heat beam 1b - $x = \frac{L}{2}$

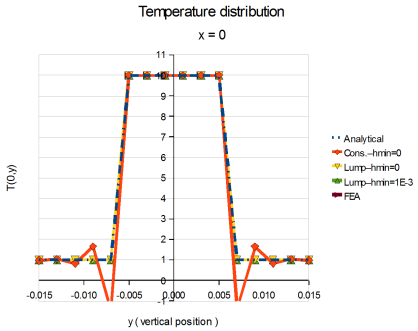


(d) Heat beam 1b - $x = L$

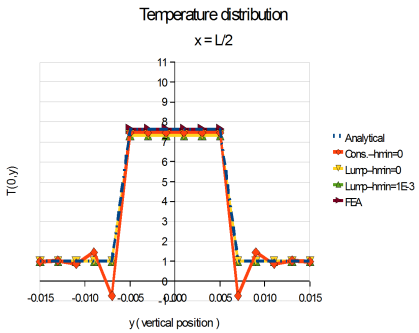


(e) Heat beam 1b - $x = L$

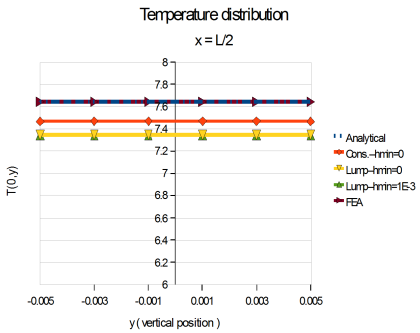
Figure 4.11 – Nodal temperature distribution for problem 1b.



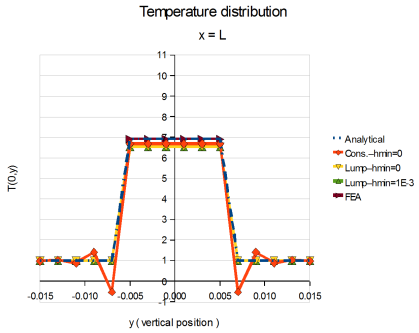
(a) Heat beam 1c - $x = 0$



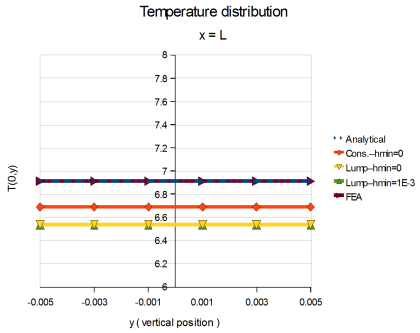
(b) Heat beam 1c - $x = \frac{L}{2}$



(c) Heat beam 1c - $x = \frac{L}{2}$

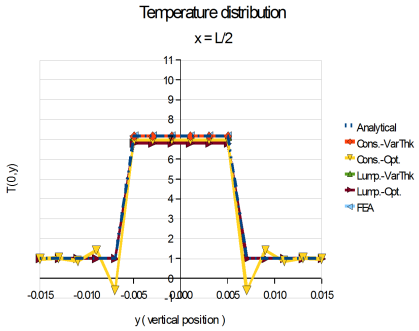


(d) Heat beam 1c - $x = L$

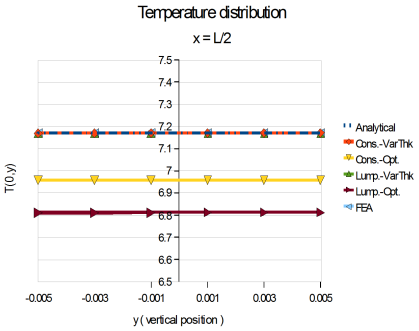


(e) Heat beam 1c - $x = L$

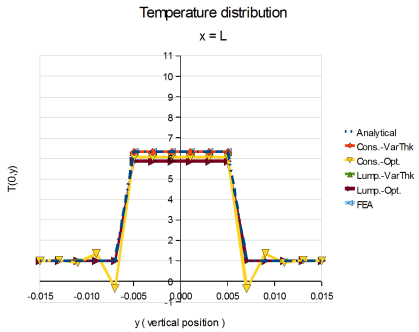
Figure 4.12 – Nodal temperature distribution for problem 1c.



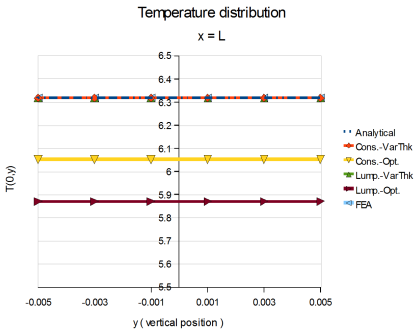
(a) Heat beam 2a - $x = \frac{L}{2}$



(b) Heat beam 2a - $x = \frac{L}{2}$

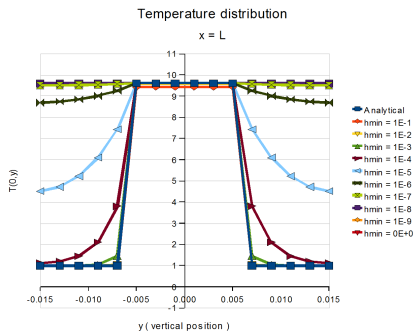


(c) Heat beam 2a - $x = L$

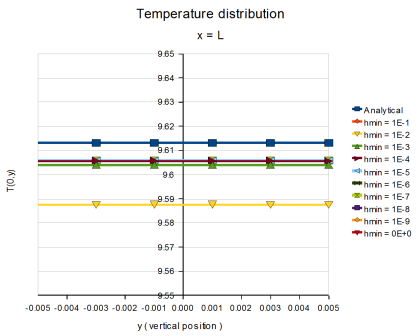


(d) Heat beam 2a - $x = L$

Figure 4.13 – Nodal temperature distribution for problem 2a.



(a) Heat beam 1b - $x = L$ - $h_{\min}$ variable



(b) Heat beam 1b - $x = L$ - $h_{\min}$ variable

Figure 4.14 – Nodal temperature distribution for problem 1b, for varying $h_{\min}$.

Part II

Comparison and confirmation of results

Chapter 5

Structural optimization

Check and comparison of results with well-known test problems from previous articles for topology optimization of purely structural problems.

5.1 MBB-beam

The test problem used, for verification of the implemented optimization relevant features, is the well-known MBB-beam. The problem is set up as seen in figure 5.1 and is the same problem as laid out by Sigmund in his comprehensive study of various filter types [12]. The design domain is defined as half of the original problem, with symmetry conditions at the cut edge, and the dimensions are 120 by 40 unit lengths and 1 unit length in thickness. Young's modulus of solid material is $E^0 = 1$, Poisson's ratio is $\nu = 0.3$, the penalisation factor is $p = 3.0$ and the minimum stiffness fraction is set to $\mu_{min} = 10^{-9}$. The imposed volume constraint is characterised by a maximum allowable volume fraction of 0.5. The design domain is discretised with two different meshes; firstly 120 by 40 square bi-linear elements as in [12] to allow for comparison, and secondly 360 by 120 elements to confirm that mesh-independence has been achieved through filtering. For the filtered solutions, a filter radius of $R = 3.5$ is imposed by the use of a sensitivity and a density filter, to allow for comparison with the results in [12].

Confirmation of OC

To confirm the implementation of the OC method and the implemented filters, the 88-line MATLAB code by Andreassen et al. [11] is used, as the original 99-line MATLAB code by Sigmund [10] doesn't have a density filter implemented. The above problem is optimized using both the MATLAB code and the implemented FORTRAN code to allow for comparison. The 88-line code is tweaked only slightly to use more strict convergence criteria than the existing code. The criteria for the bisection scheme is changed to 10^{-4}.

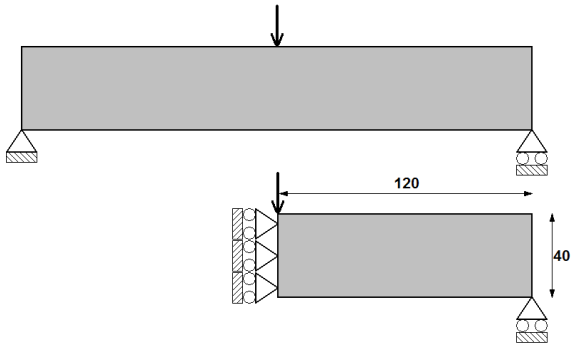


Figure 5.1 – The problem setup for the MBB beam.

No filtering

The solutions don't converge, but start to oscillate around a value of the objective function and the solutions don't change significantly after about the 60th iteration. Therefore, figure 5.2 shows the solutions obtained after 100 iterations. As can be seen, the solutions are exactly the same, confirming that the OC method has been implemented correctly in the FORTRAN programme. The appearance of checkerboards is noted as a reason to apply filtering.

(a) FORTRAN programme **(b)** 88-line MATLAB

Figure 5.2 – 120x40 - No filtering. Solver; OC Compliance, it.; **(a)** $f_0 = 202.1597$, 100 - **(b)** $f_0 = 202.1597$, 100

Sensitivity filtering

As can be seen from figures 5.3, the sensitivity filter implementation gives aesthetically the exact same solution as the 88-line MATLAB code. The values of the objective function are also exactly the same during the iteration process, however, due to slight implementation differences the MATLAB code stops one iteration earlier. As expected the filtering results in a worse result than the unfiltered result with respect to the objective function, but now has more physically relevant boundaries. When comparing figure 5.3 and the corresponding objective function values, with results obtained for an equivalent sensitivity filter in [12], it is seen that the same results are obtained both quantitatively and qualitatively.

(a) FORTRAN programme **(b)** 88-line MATLAB

Figure 5.3 – 120x40 - Sensitivity filtering, filter radius; R = 3.5
Solver; OC - Compliance, it.; **(a)** $f_0 = 210.0167$, 91 - **(b)** $f_0 = 210.0232$, 90

Density filtering

As can be seen from figure 5.4, the density filter produces almost the same results for both the FORTRAN programme and MATLAB code. Due to implementation differences, there is a slight difference in the final obtained compliance of the solutions. When comparing the results to those obtained for an equivalent density filter in [12], the aesthetics of the solutions are much alike but the OC method used here doesn't produce the exact same solutions as the MMA algorithm used by Sigmund.

(a) FORTRAN programme **(b)** 88-line MATLAB

Figure 5.4 – 120x40 - Density filtering, filter radius; R = 3.5
Solver; OC - Compliance, it.; **(a)** $f_0 = 220.7074$, 621 - **(b)** $f_0 = 220.5316$, 629

Mesh-independence

Due to the extensive computation time it takes for the OC method to solve the problem with an applied density-filter, this was only run for a slightly finer mesh (150x50) than the original mesh. Figure 5.5 shows that a there seems to be a problem with obtaining a mesh-independent solution using the density filter. This could very well be due to the density-filter having a slightly smaller length-scale than the sensitivity-filter. As the OC method is only used for the start of the project, this is seen as insignificant, but could maybe be solved using a continuation approach. The OC method takes a much longer time than the GCMMA algorithm, when used with a density-filter as the OC method calls the density-filter for each function call in the bisection scheme.

(a) Sensitivity filtering - 360x120

(b) Density filtering - 150x50

Figure 5.5 – Filter radius; R = 3.5 - FORTRAN programme - Solver; OC Compliance, it.; **(a)** $f_0 = 213.51839$, 100 - **(b)** $f_0 = 222.9286$, 303

Confirmation of GCMMA

As it is not possible to directly compare the results for the GCMMA implementation to those of the MATLAB code, the final solutions are simply qualitatively and quantitatively compared to the confirmed results for the OC method and the results reached in [12].

No filtering

As figure 5.6 shows, the GCMMA algorithm produces a different solution than the OC method used in figure 5.2. The solution is better with respect to the objective function, and is closer to the value obtained in [12]. The aesthetics of the solution is also much more similar to the solution produced by the MMA algorithm, which is to be expected as MMA and GCMMA are closely related, whereas the OC method is purely heuristic. Again the abundance of checkerboards is noted.

Figure 5.6 – 120x40 - No filtering - FORTRAN - Solver; GCMMA Compliance, it.; $f_0 = 196.0788$, 92 (94)

Density filtering

It can be seen that figure 5.7 much resembles the corresponding solution for OC, figure 5.4. The reached compliance is slightly better, and closer to the value obtained in [12] as expected.

Mesh-independence

As figure 5.8 shows, the same solution is reached for the refined mesh when the density filter is active. The optimization was stopped after 1000 itera-

Figure 5.7 – 120x40 - Density filtering, filter radius; R = 3.5 - FORTRAN Solver; GCMMA - Compliance, it.; $f_0 = 218.1547$, 887 (890)

tions, and this is considered acceptable as the solution was close to convergence.

Figure 5.8 – 360x120 - Filter radius; R = 3.5 - Fortran programme Solver; GCMMA - Compliance, it.; $f_0 = 222.1601$, 1000 (1011)

Based on the observations made in this section, it can be concluded that the OC method, GCMMA algorithm and the two filters have been implemented correctly in the FORTRAN programme for compliance optimization wrt. elastostatic stiffness optimization.

Chapter 6

Thermal optimization

Check and comparison with results from previous articles for topology optimization of purely thermal problems.

6.1 Distributed heating plate

The test problem is set up as in figure 6.1, and is a 20 unit length square plate, with a thickness of 1 unit length, subjected to evenly distributed heating all over the plate - that is a constant heat flux applied to all nodes. At the middle of the left-hand side there is a 2 unit length long heat sink, where the nodal temperature values are set to zero. The thermal conductivity of solid material is $k^0 = 1$, the penalisation factor is $p = 3.0$ and the minimum thermal stiffness fraction is set to $\mu_{min} = 10^{-3}$. The imposed volume constraint is characterised by a maximum allowable volume fraction of 0.3. The design domain is discretised by 3 different meshes of various fineness using square bi-linear elements. Firstly, a 20x20 element mesh is used to compare results with the 91-line modified MATLAB code, second and thirdly 40x40 and 80x80 meshes are used to show mesh-independence. A total heat flux of 4.38 is applied over the plate; equivalent to 0.001 at all active nodes for the 20x20 discretisation.

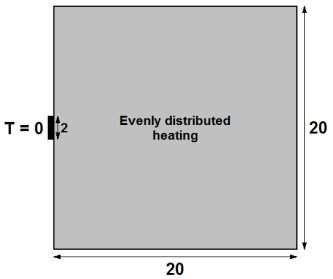


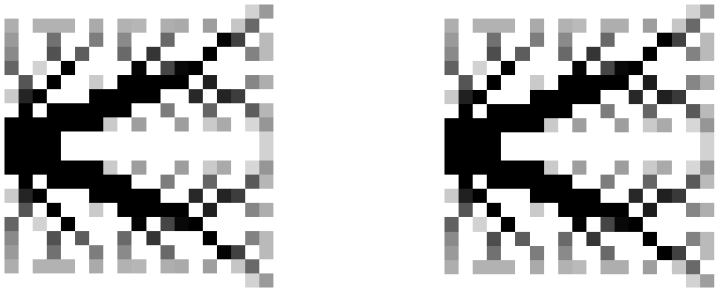
Figure 6.1 – The problem setup for the distributed heating plate.

Confirmation of OC

To confirm the implementation of OC, the 99-line MATLAB code by Sigmund [10] is modified for heat conduction problems by following the changes specified in app. 5.1.6 of Bendsøe and Sigmunds book [24]. The above problem is optimized using both the modified MATLAB code and the implemented FORTRAN programme.

No filtering

As figure 6.2 shows, the solution obtained with the FORTRAN programme is the same as obtained with the 91-line modified MATLAB code, except for the extra iteration in the FORTRAN programme, as noted in section 5.1.



(a) FORTRAN programme

(b) 91-line MATLAB

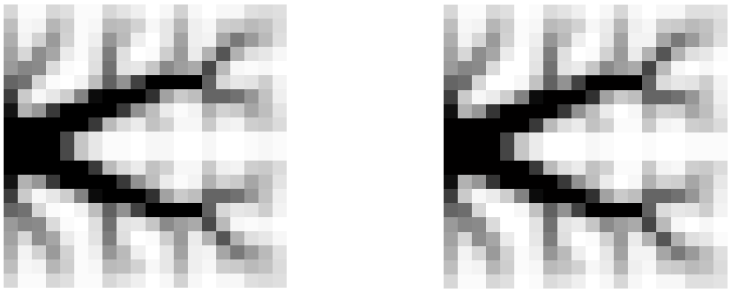
Figure 6.2 – 20x20 - No filtering - Solver; OC
Compliance, it.; **(a)** $f_0 = 45.8196$, 39 - **(b)** $f_0 = 45.8211$, 38

Sensitivity filtering

As can be seen from figures 6.3, the sensitivity filter implementation also gives exactly the same solution as the 91-line MATLAB code except for the extra iteration.

Density filtering

As the 91-line modified MATLAB code doesn't have a density filter implemented, the results obtained using the density filter implemented in the FORTRAN programme are compared to those obtained with the sensitivity filter. When comparing figures 6.3 and 6.4, it can be seen that the solutions differ slightly, but still overall look alike. The design is slightly better wrt. thermal compliance, but nothing of significance.



(a) FORTRAN programme

(b) 91-line MATLAB

Figure 6.3 – 20x20 - Sensitivity filtering, filter radius; R = 1.2 - Solver; OC Compliance, it.; **(a)** $f_0 = 60.7943$, 42 - **(b)** $f_0 = 60.8525$, 41

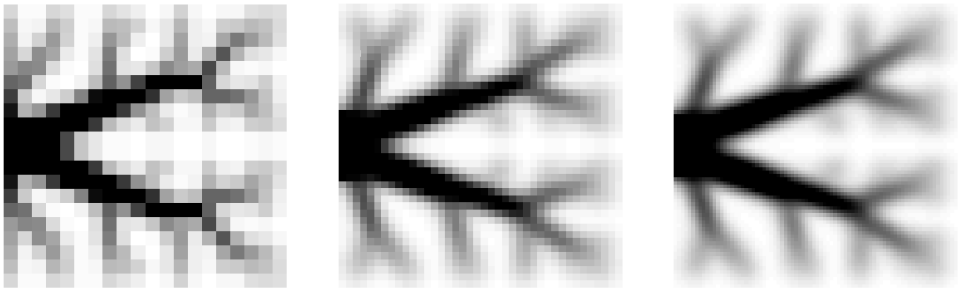
Figure 6.4 – 20x20 - Density filtering, filter radius; R = 1.2 FORTRAN - Solver; OC - Compliance, it.; $f_0 = 60.0206$, 98

Mesh-independence

Figures 6.5 and 6.6 show that mesh-independence is obtained using both the implemented filters. It can be seen that the thermal compliance of the solution using density filtering converges from above with reduction of element-size, but for the sensitivity filtered solution no such trend is seen. Of course, the problem would need to be run for several finer discretisations, before a conclusion can be formed on this - but that is outside the scope of this project. It is noted, that it appears the density filter has a slightly shorter length-scale than the sensitivity filter, as the "branches" of the solution appear more compact and less blurred.

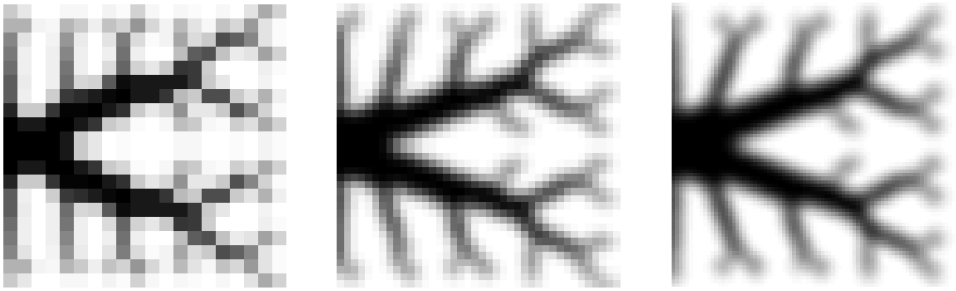
Confirmation of GCMMA

It is not possible to directly compare the results obtained from the implemented GCMMA algorithm to anything, other than the results from the OC method. This should also be sufficient, as the thermal implementation of the GCMMA algorithm is analogous with the implementation for static mechanical problems, which has already been verified in section 5.1.



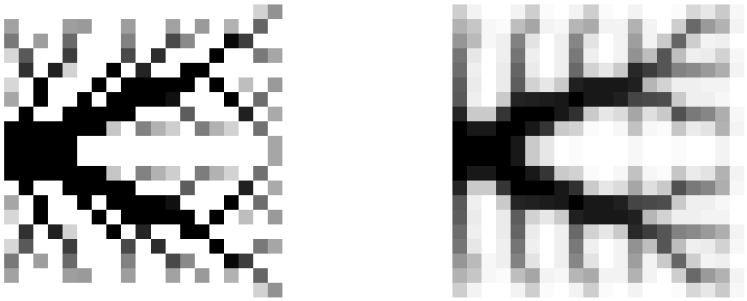
(a) 20x20 **(b)** 40x40 **(c)** 80x80

Figure 6.5 – Sensitivity filtering, filter radius; R = 1.2 - FORTRAN - Solver; OC Compliance; **(a)** $f_0 = 60.7943$, 42 - **(b)** $f_0 = 62.4461$, 47 - **(c)** $f_0 = 61.2299$, 46



(a) 20x20 **(b)** 40x40 **(c)** 80x80

Figure 6.6 – Density filtering, filter radius; R = 1.2 - FORTRAN - Solver; OC Compliance; **(a)** $f_0 = 60.0206$, 98 - **(b)** $f_0 = 58.6866$, 297 - **(c)** $f_0 = 57.7567$, 286



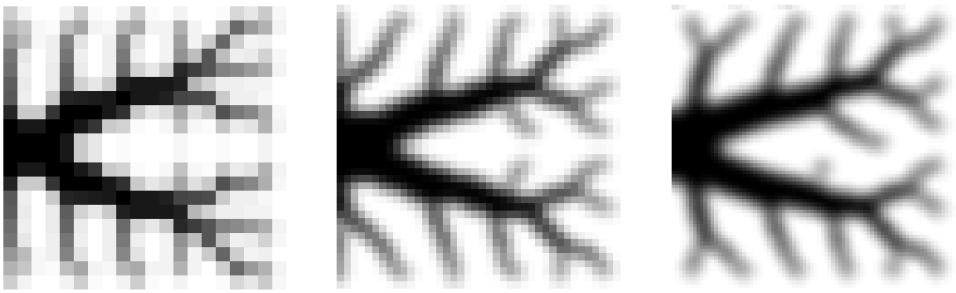
(a) No filter; R = 0.0 **(b)** Density filtering; R = 1.2

Figure 6.7 – 20x20 - FORTRAN - Solver; GCMMA Compliance, it.; **(a)** $f_0 = 46.5680$, 54 (66) - **(b)** $f_0 = 60.1866$, 66 (78)

Comparing figures 6.7a and 6.7b with figures 6.2 and 6.4 respectively, it can be seen that the GCMMA algorithm produces qualitatively almost the same solutions as the OC method. The thermal compliance of the solutions are also seen to be very close.

Mesh-independence

Figure 6.8 does not give as clear evidence towards confirming that mesh-independence has been achieved through the density filtering for the GCMMA algorithm. It is, however, hard to prove mesh-independence due to the many local minima that can be found. Using a continuation method, where the penalisation factor is increased gradually after convergence, could help to ensure getting the same optimal solution for each mesh. This is, however, not the prime purpose of this project, so it is left as is.



(a) 20x20 **(b)** 40x40 **(c)** 80x80

Figure 6.8 – Density filtering, filter radius; R = 1.2 - Solver; GCMMA Compliance, it.; **(a)** $f_0 = 60.1866$, 66 (78) - **(b)** $f_0 = 58.4090$, 316 (465) - **(c)** $f_0 = 56.9295$, 1442 (1453)

Based on the observations made in this section, it can be concluded that the OC method, GCMMA algorithm and the two filters have been implemented correctly in the FORTRAN programme for thermal compliance optimization with respect to steady-state heat transfer. It is also noted that generally the solutions obtained for thermal optimization are less prone to pure 0-1 designs as compared to the mechanical optimization.

Chapter 7

Coupled optimization

Check and comparison of results with well-known test problems from previous articles for topology optimization of coupled thermal and structural problems.

7.1 Thermal actuator

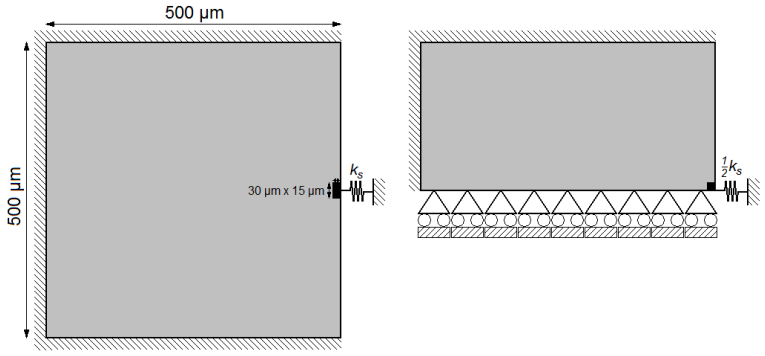


Figure 7.1 – The problem setup for the thermal actuator.

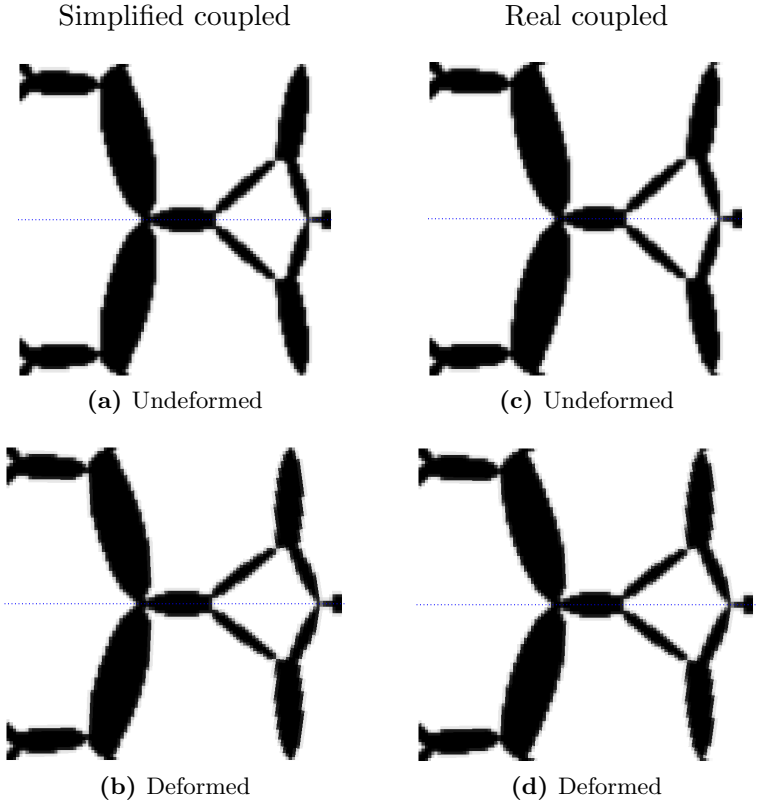
The first test problem used for verification of the implemented coupled thermomechanical is the design of thermal micro-actuators. The problem setup is shown in figure 7.1 and is the same problem as laid out by Sigmund in [6]. The design domain is defined as half of the original problem, with symmetry conditions at the cut edge, and the dimensions are 500 by 250 µm and a thickness of 15 µm. The objective function is defined as $\Phi = -\mathbf{lu}$, where l is a vector with a 1 at the DOF that is wished to be optimized. In this case, the x-displacement of the bottom right node is to be maximised. A constant temperature change of 100 K throughout the domain is applied by "hard-coding", bypassing the need to solve the temperature distribution and simplifying the adjoint sensitivity for this problem to:

$$\frac{\partial \Phi}{\partial \rho_e} = \boldsymbol{\lambda}^T \left(\frac{\partial \mathbf{f}}{\partial \rho_e} - \frac{\partial \mathbf{K}}{\partial \rho_e} \mathbf{u} \right) \tag{7.1}$$

with $\boldsymbol{\lambda}$ as the solution to the adjoint problem:

$$\mathbf{K}\boldsymbol{\lambda} = -\mathbf{l} \tag{7.2}$$

Young's modulus of solid material is $E^0 = 200\text{GPa}$, Poisson's ratio is $\nu = 0.31$, the thermal expansion coefficient is $\alpha = 15 \times 10^{-6}\text{K}^{-1}$ and the minimum stiffness fraction is set to $\mu_{min} = 10^{-3}$. The imposed volume constraint is characterised by a maximum allowable volume fraction of 0.25. The design domain is discretised using 100 by 50 square bi-linear elements as in [6] to allow for comparison, and the stiffness of the spring is set to $k_s = 2000\text{Nm}^{-1}$. The filter radius is set to 1.2 times the element size, that is 6 μm, to prevent checkerboards, but still allowing thin "pivot-points" to form. The problem is solved for $p = 3.0$ and the convergence criteria for the GCMMA algorithm is set to 10^{-3}.



(a) Undeformed **(c)** Undeformed

(b) Deformed **(d)** Deformed

Figure 7.2 – 100x50 - Nodal output, it.;
(b) $u_{\text{out}} = 3.4836\mu\text{m}$, 239 (254) - **(d)** $u_{\text{out}} = 3.4777\mu\text{m}$, 266 (276)

By comparison, figure 7.2a somewhat resembles the layout of the optimal solution found by Sigmund in [6]. The optimal design found here, looks like

a combination of the two solutions for the 2000Nm^{-1} and the 200Nm^{-1} springs found by Sigmund. The objective function, however, is very close to the $\sim$3.4μm found by Sigmund for the 2000Nm^{-1} spring. The differences that appear in the optimal solutions, could be due to many things; the GCMMA algorithm has been used instead of the MMA algorithm, and therefore a density filter has been used instead of a sensitivity filter, and also a continuation scheme has not been used.

Coupled optimization

Now the 100 K temperature rise is applied through the temperature field, so a real coupled optimization of the same problem is run to check if the implementation of this is correct. The sensitivity of the objective function used is as found in section 3.1. As figure 7.2c shows, almost the exact same solution is reached through the coupled analysis. The slight differences between the coupled and simplified solutions are attributed to slight implementation differences and various numerical irregularity, like accumulated errors from the additional thermal simulation.

7.2 Coupled test problem - Rodrigues A

The first test problem for coupled optimization wrt. to mechanical compliance (as described in appendix B.1) is inspired by an old article by Rodrigues and Fernandes [7]. But due to all material parameters not being specified the problem has not been reproduced directly.

As seen in figure 7.3, the design domain has the dimensions 0.72m by 0.48m with a thickness of 0.01m. The domain is fully clamped on both the left- and right-hand sides, where there is a 2-element thick non-design area imposed. Young's modulus of solid material is $E^0 = 210$GPa, Poisson's ratio is $\nu = 0.30$, the thermal expansion coefficient is $\alpha = 15 \times 10^{-6} K^{-1}$ and the minimum stiffness fraction is set to $\mu_{min} = 10^{-9}$. Firstly, the problem is discretised by 60x30 elements, as in [7], and solved using GCMMA with a penalisation factor $p = 3.0$ and a convergence criteria of 10^{-2}. Secondly, the problem is discretised by 120x60 elements and solved using GCMMA and a continuation scheme, where the problem is solved until convergence for an increasing penalisation factor from 1.0 to 3.0 over 8 steps. A filter radius of 1.2 times the element size is also imposed to prevent checkerboards. The concentrated force is applied at the middle bottom node and has the value 10kN and the imposed volume constraint is characterised by a maximum allowable volume fraction of 0.40. The problem is solved for three different evenly distributed temperature rises over the entire domain.

Figure 7.4 shows that the obtained solutions are qualitatively very close to those of Rodrigues and Fernandes, when using neither a continuation

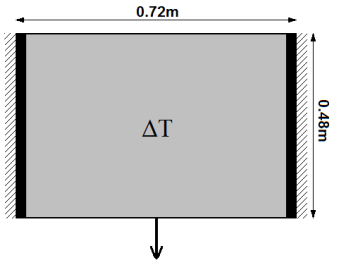


Figure 7.3 – The problem setup for Rodrigues A.

(a) $\Delta T = 0$ **(b)** $\Delta T = 1\text{K}$ **(c)** $\Delta T = 4\text{K}$

Figure 7.4 – No filtering - Coupled optimization wrt. mechanical compliance. Comp., vol.frac., it.; **(a)** $f_0 = 0.2732$, 0.400, 60 (67) - **(b)** $f_0 = 0.5122$, 0.400, 50 (62) - **(c)** $f_0 = 1.5798$, 0.335, 57 (186)

(a) $\Delta T = 0$ **(b)** $\Delta T = 1\text{K}$ **(c)** $\Delta T = 4\text{K}$

Figure 7.5 – Finer mesh - Dens. filtering w. cont. method Comp., vol.frac., it.; **(a)** $f_0 = 0.2911$, 0.400, 1000 (1106) - **(b)** $f_0 = 0.4499$, 0.400, 373 (517) - **(c)** $f_0 = 1.4198$, 0.270, 117 (462)

method nor filtering. However, due to many of the parameters being unknown and the differences in solution method and implementation, the compliance measures are in no way comparable. The results, however, still show the influence that a temperature rise has on the optimised solution. As the temperature rise is increased, it can be seen that the solution changes considerably and the obtained compliance worsens. It is also noted that when the temperature rise is increased sufficiently, as seen in figures 7.4c and 7.5c, the used volume fraction starts to decrease. This is due to the fact that now the more material that is in use, larger thermal strains will arise and work against the compliance minimisation of the system. The blurry non-0-1 design is due to the penalisation not "working" when the volume-constraint is not active.

The appearance of extra arms in figure 7.5a when compared to figure 7.4a, is likely due to finer mesh bringing with it the possibility of finer details. Also, more material is available to be used within the designable domain, as the fixed area is not as wide. It is, however, noted that the compliance is larger for the finer discretisation, even though a continuation scheme is used. This could be due to the latter being a local minimum. The differences between figures 7.4b and 7.5b are also attributed to these things.

When compared to the results obtained by Rodrigues [7], a lot of similarities are seen and the differences can be attributed to the different optimization algorithms, differences in parameters and that Rodrigues uses nine-node elements for some of his results, whereas only four-node elements are used here.

7.3 Coupled test problem - Rodrigues B

The second test problem for coupled optimization wrt. to mechanical compliance is inspired by the same article by Rodrigues and Fernandes [7] as the previous test problem. Again due to all material parameters not being specified in the article the problem has not been reproduced directly.

Figure 7.6 shows the design domain, which has the dimensions 0.50m by 0.28m with a thickness of 0.01m, which is discretised by 100x56 square bilinear elements. Like the previous test problem, the domain is fully clamped on both the left- and right-hand sides, and the various parameters are the same as before - also for the continuation scheme and filter radius. The imposed volume constraint is characterised by a maximum allowable volume fraction of 0.33. The total applied force is 15000N distributed on the top edge. The problem is solved for two different evenly distributed temperature rises over the entire domain, and lastly where 0°C and 10°C is imposed on the left- and right-hand sides respectively.

It can be seen that the solutions for the evenly distributed temperatures are qualitatively almost identical. The compliance for the $\Delta T = 10$K case is, however, slightly smaller than for the 0 K case. It, therefore, appears that the thermal strains act against the upwards pulling forces along the upper

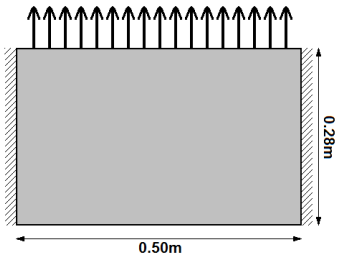


Figure 7.6 – The problem setup for Rodrigues B.

(a) $\Delta T = 0$ **(b)** $\Delta T = 10\text{K}$ **(c)** $\Delta T = 0 - 10\text{K}$

Figure 7.7 – Coupled optimization wrt. mechanical compliance. Comp., vol.frac., it.; **(a)** $f_0 = 2.276$, 0.330, 1408 (1571) - **(b)** $f_0 = 2.237$, 0.330, 1154 (1342) - **(c)** $f_0 = 2.819$, 0.330, 319 (664)

boundary, causing less displacement and thereby a smaller compliance. It is also noted that the solutions are not symmetrical, as would be expected for the perfectly symmetrical problem. But this is thought to be due to the inaccuracies in the nodal coordinates of the input file (i.e. 0.279999994 instead of 0.28) - this has been taken into account for all later studies.

Figure 7.7c shows that a varying temperature field, however, does change the design considerably. It is seen that the design is close to the same in the cold left-side of the domain, but in the warm right-side the curved beam construction has moved up considerably. This is most likely because this causes the thermal expansion of the right-side to counteract the pulling of the vertical forces.

The similarities with the solutions obtained by Rodrigues are noted, but like in section 7.2 most of the parameters are unknown and the solution method is different, so the compliance measures are in no way comparable.

Part III

Topology optimization of convection problems

Chapter 8

Purely thermal test problem

8.1 Heated plate affected by convection

Optimization wrt. nodal temperature

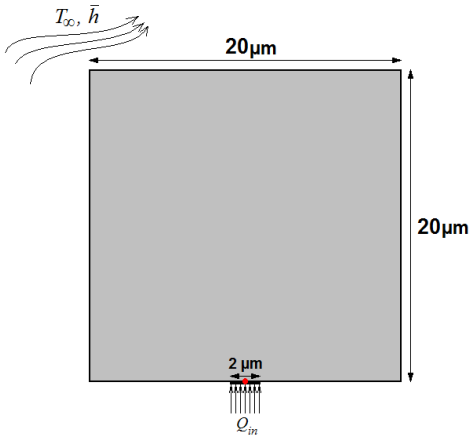
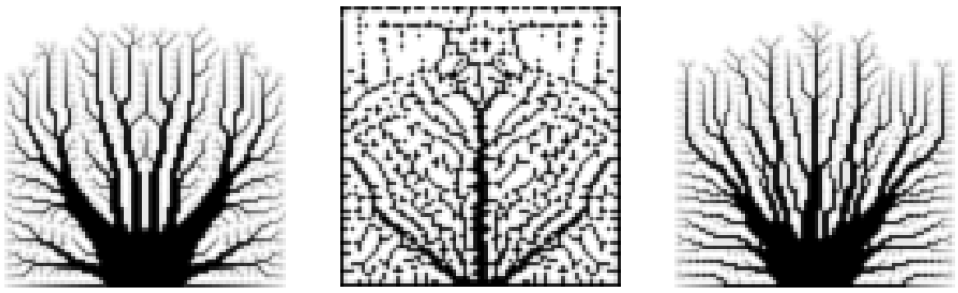


Figure 8.1 – Setup for the heated plate affected by convection. Point a is marked by the red dot.

The test problem is set up as in figure 8.1, and is very similar to the problem examined in section 6. It is a 20 μm square plate, with a thickness of 1 μm, subjected to a variation of top, bottom and side convection. A total heat flux of $Q_{in} = 3\,\mu\text{W}$ is imposed along 2μm at the middle of the lower side. The domain is discretised by 80x80 square bilinear elements. The optimization is carried out for the minimisation of the nodal temperature at point A, that is the transport of energy away from point A must be optimized.

The thermal conductivity of solid material is $k^0 = 43.0\text{Wm}^{-1}\text{K}^{-1}$, the penalisation factor is $p = 3.0$ and the minimum thermal stiffness fraction is set to $\mu_{min} = 10^{-9}$. The surrounding fluid has temperature of $T_\infty = 0°\text{C}$, a

nominal convection heat transfer coefficient of $\bar{h} = 2500\,\mathrm{Wm^{-2}K^{-1}}$ (a large value, but this allows the desired observable effects to show themselves) and the minimum convection fraction is set to $h_{\min} = 10^{-3}$. The imposed volume constraint is characterised by a maximum allowable volume fraction of 0.3, and a radius of 1.2 times the element size is used to remove checkerboards without imposing a length-scale of significance.

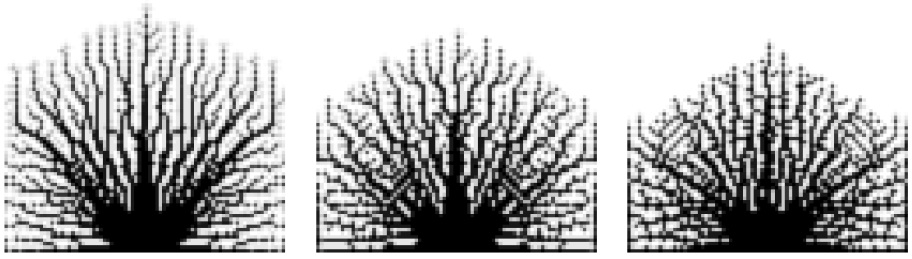


(a) TBC **(b)** ISC, BSC **(c)** TBC, ISC, BSC

Figure 8.2 – Temperature, it.;
(a) $T_a = 2.8484$, 197 (331) - **(b)** $T_a = 17.8598$, 140 (143)
(c) $T_a = 2.7521$, 177 (262)

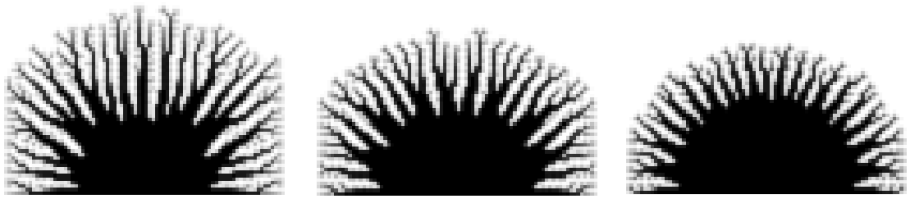
Figure 8.2 shows the solutions for the problem solved with a variation of convection loads. Comparing subfigures 8.2a and 8.2c, the effect of including SC compared to TBC only can be seen, and the result is as expected. The optimization algorithm appears to takes advantage of the fact that, the total area affected by convection now includes the sides, and so the final solution has more thin arms spreading out from the heat source. Figure 9.2b shows the solution obtained for the problem run with SC only. It can be seen that the solution has many thin arms running through the entire domain, as increasing the side area is the only way of getting rid of the energy efficiently. To further explore the effect of SC compared to TBC, the nominal convection heat transfer coefficient of the side convection (ISC and BSC) is increased several factors of the top and bottom convection equivalent. The solutions can be seen in figure 8.3 and again the results are as expected. It can be seen that as the magnitude of the side convection increases, the material is collected closer to the heat source. This makes perfect sense, as the heat is removed more efficiently and the heat, therefore, does not travel as far into the domain.

When the top and bottom convection is increased compared to the side convection, as in figure 8.4, the effect of the material being collected closer to the heat source is again observed. Also, as the magnitude of TBC increases compared to that of SC, it can be seen that less thin arms reaching out appear in the optimal solution. This is due to the fact that, in this case it is more effective to increase the planar surface area, as it is here the heat is dissipated effectively.



(a) $\bar{h}_{SC} = 5\bar{h}_{TBC}$ **(b)** $\bar{h}_{SC} = 10\bar{h}_{TBC}$ **(c)** $\bar{h}_{SC} = 20\bar{h}_{TBC}$

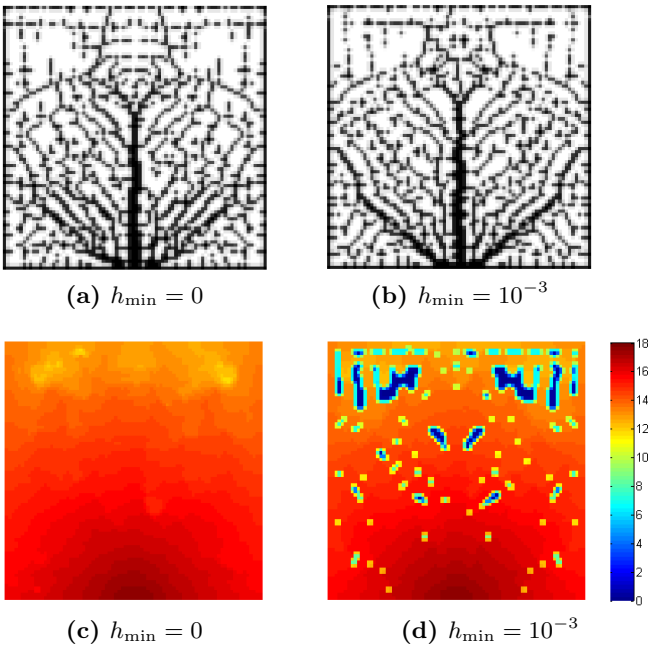
Figure 8.3 – Temperature, it.;
(a) $T_a = 2.3690$, 168 (199) - **(b)** $T_a = 2.0435$, 178 (202)
(c) $T_a = 1.6464$, 129 (144)



(a) $\bar{h}_{TBC} = 5\bar{h}_{SC}$ **(b)** $\bar{h}_{TBC} = 10\bar{h}_{SC}$ **(c)** $\bar{h}_{TBC} = 20\bar{h}_{SC}$

Figure 8.4 – Temperature, it.;
(a) $T_a = 1.1713$, 152 (184) - **(b)** $T_a = 0.8802$, 174 (205)
(c) $T_a = 0.7012$, 184 (207)

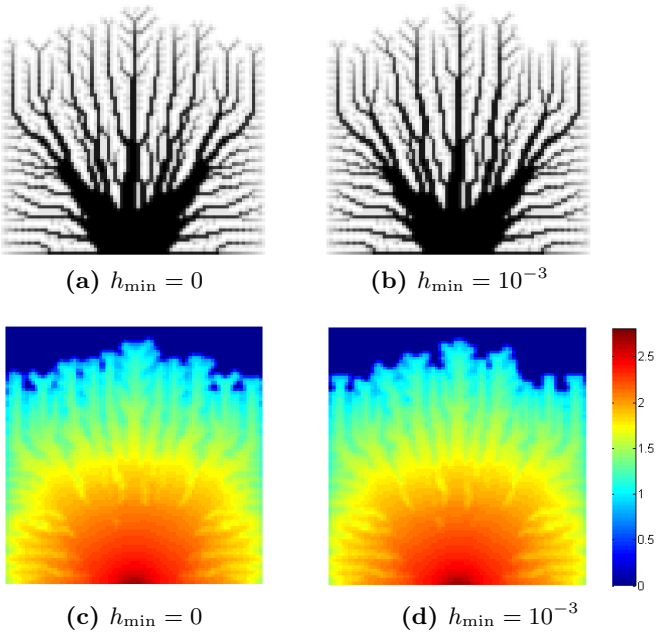
The optimizations have all been run with a minimum convection contribution, because this was found to help model the temperature distribution better in the analytical study in section 4.4. However, to see if this really makes a difference to the obtained solutions, the problem was also run for all-around convection and also SC only, with $h_{\min} = 0$. As can be seen in figure 8.5, the obtained solutions are definitely different, but bear many similarities. The important thing to recognize is the significant difference in the temperature distributions for the two, it can be seen that fluid temperature is obtained in the largest of the 0-density areas. It is contemplated that, although it has not made a huge difference in the obtained solutions and objective functions, modelling the temperature distribution correctly is hugely important, especially for the coupled optimization where the temperature field starts to affect the displacement field of the model. However, when looking at figure 8.6, the differences are not significant and hard to spot. This is most likely due to the fact that, as discussed earlier, the magnitude of the TBC contribution is much larger than the SC contribution and, therefore, any differences due to the SC drown out.



(a) $h_{\min} = 0$ **(b)** $h_{\min} = 10^{-3}$

(c) $h_{\min} = 0$ **(d)** $h_{\min} = 10^{-3}$

Figure 8.5 – ISC and BSC only - Optimised solutions and temp. distributions Temperature, it.; **(a)** $T_a = 17.8945$, 151 (152) - **(b)** $T_a = 17.8598$, 140 (143)



(a) $h_{\min} = 0$ **(b)** $h_{\min} = 10^{-3}$

(c) $h_{\min} = 0$ **(d)** $h_{\min} = 10^{-3}$

Figure 8.6 – TBC, ISC and BSC - Optimised solutions and temp. distributions Temperature, it.; **(a)** $T_a = 2.7514$, 246 (343) - **(b)** $T_a = 2.7521$, 177 (262)

Crosschecks

To check if the above conclusions about the optimizations are correct, a crosscheck is performed for the problem with varying convection (figure 8.2) and also the problem where the magnitude of ISC and TBC are varied (figures 8.3 and 8.4 respectively). The results are seen in tables 8.1 to 8.3, where coloured text indicates the lowest (and therefore optimal) temperature for the given analysis. Blue text indicates solutions that fit the expected trend and red text indicates solutions that don't fit the expected trend. It is seen that most of the results are as expected, the lowest temperature is obtained by the optimized distributions for a given analysis. However, there are two anomalies where other designs perform better than the solutions returned by the optimization. This is most likely due to the optimization finding local minima, instead of the global optimal minimum. This could be checked again by applying a continuation approach, but unfortunately time has not allowed for this to be done.

		Distribution		
		TBC	TBC, SC	SC
Analysis	TBC	2.8484	2.8782	4.0716
	TBC, SC	2.7386	2.7521	3.8596
	SC	28.758	26.740	17.860

Table 8.1 – Crosscheck for the problems in figure 8.2.

		Distribution		
		$5\bar{h}$	$10\bar{h}$	$20\bar{h}$
Analysis	$5\bar{h}$	2.3690	2.4303	2.4934
	$10\bar{h}$	2.0197	2.0434	2.0696
	$20\bar{h}$	1.6484	1.6475	1.6464

Table 8.2 – Crosscheck for the problems in figure 8.3.

		Distribution		
		$5\bar{h}$	$10\bar{h}$	$20\bar{h}$
Analysis	$5\bar{h}$	1.1712	1.1901	1.2329
	$10\bar{h}$	0.8836	0.8802	0.8904
	$20\bar{h}$	0.7096	0.7020	0.7012

Table 8.3 – Crosscheck for the problems in figure 8.5.

Chapter 9

Thermomechanical test problem

9.1 Thermal actuator affected by convection

To see the effect of the modelled convection on a coupled thermomechanical system, a thermal actuator problem is set up as in figure 9.1. The stiffness of the added spring is $k_s = 10^5 \mathrm{Nm}^{-1}$. The temperature of the clamped boundary is fixed to 10°C, and the domain is subjected to a variation of convection, where the fluid temperature is set to either $T_\infty = 1°\mathrm{C}$ or $T_\infty = 20°\mathrm{C}$, the nominal convection heat transfer coefficient is $\bar{h} = 25 \mathrm{Wm}^{-2}\mathrm{K}^{-1}$ and the minimum convection fraction is set to $h_{\min} = 10^{-3}$. The thermal conductivity of solid material is $k^0 = 43.0 \mathrm{Wm}^{-1}\mathrm{K}^{-1}$ and the minimum stiffness fraction is set to $\mu_{min} = 10^{-9}$. The imposed volume constraint is characterised by a maximum allowable volume fraction of 0.3 or 0.5 as specified, and a radius of 1.2 times the element size is used to remove checkerboards without imposing a length-scale of significance. The objective function is to maximise the nodal displacement, of the lower rightmost node, in the downwards direction. The sensitivities are, therefore, as laid out in section 3.1.

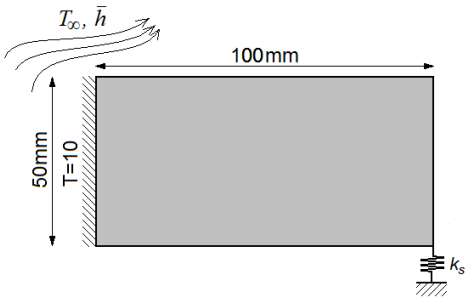


Figure 9.1 – Setup for the thermal actuator affected by convection.

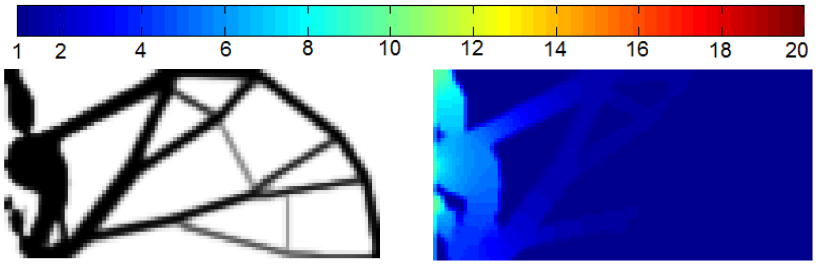
Figure 9.2 shows the obtained solution for TBC, ISC and BSC with the fluid temperature set to 1°C. The temperature can be seen to have been lowered substantially for most of the domain, and the actuating mechanism is concentrated at the area right up to the heated and clamped boundary. The rest of the solution is pretty much unaffected by the thermal expansion, and is merely a truss-like structure that can withstand the upwards-force of the spring.

When the fluid temperature is changed to 20°C, as in figure 9.3, it is again seen that the temperature reaches the fluid temperature for much of the domain. The actuating mechanism is still concentrated at the area right up to the clamped boundary, but the mechanism is considerably different. It can clearly be seen that small "fingers" have formed at the top actuating arm, to draw in as much heat from the surrounding fluid as possible, causing this part to expand and thereby twist the rest of the structure around the lower pivot to maximise the nodal-output.

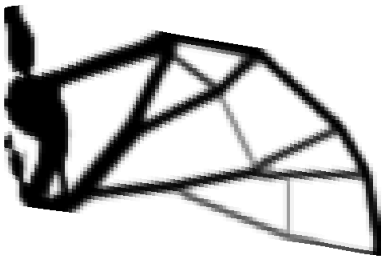
Allowing the solution to use more of the design-domain, that is raising the volume constraint to 50%, gives the result shown in figure 9.4. It is seen that a lot of the extra usable material is concentrated at the bottom of the truss-like structure, but also used to make the fingers of the top actuating arm longer. The longer fingers help drawn in heat where it is needed.

In the hope of seeing even more significant effects of the design-dependent convection that has been implemented, the optimization is run for the problem with only ISC and BSC applied. The obtained solutions can be seen in figures 9.5 and 9.6. The temperature levels in the structure can be seen to be farther from the fluid temperature, as compared to the solutions with TBC also, for both cases. This is as expected, as the side convection is active over a much smaller area compared to TBC. The solution for 1°C fluid temperature can be seen to be quite a bit different from the equivalent solution in figure 9.2. It can be seen that there is only one actuator "arm" and a single pivot-point at boundary, contrary to the two from before. The solution for 20°C fluid temperature, bears quite the resemblance to the equivalent solution in figure 9.3, but the truss-like structure has many more thin bars. This makes sense as the area along the side of the structure now needs to be maximised to increase the absorption of energy from the surrounding fluid. It also appears that the actuating mechanism has a more complex structure, in that it has a few extra joints/pivot-points.

When raising the allowable volume-fraction to 0.5, it can be seen in figure 9.7 that an almost web-like structure is obtained. The many small bars and holes, fit nicely with the fact that the side area of the construction should be maximised to draw in as much heat from the surrounding fluid as possible.

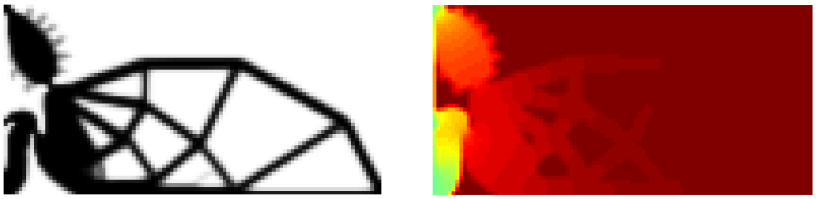


(a) Undeformed solution **(b)** Temperature distribution

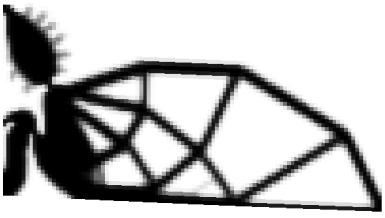


(c) Deformed solution ($u_y \times 10^3$)

Figure 9.2 – $T_\infty = 1°C$ - TBC, ISC and BSC - Vol. frac. = 0.3 Nodal displacement, it.; **(a)** $u_{\text{out}} = -1.5790\mu\text{m}$, 132 (1347)

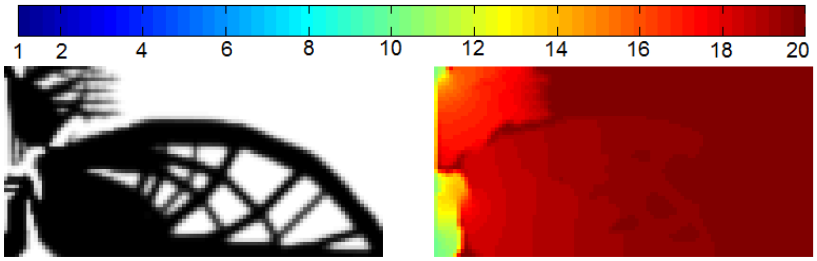


(a) Undeformed solution **(b)** Temperature distribution



(c) Deformed solution ($u_y \times 10^2$)

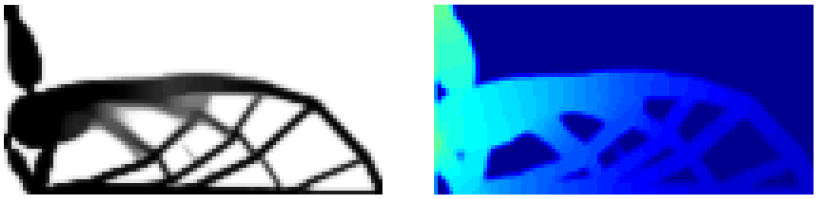
Figure 9.3 – $T_\infty = 20°C$ - TBC, ISC and BSC - Vol. frac. = 0.3 Nodal displacement, it.; **(a)** $u_{\text{out}} = -4.2871\mu\text{m}$, 653 (1939)



(a) Undeformed solution **(b)** Temperature distribution

(c) Deformed solution
($u_y \times 10^2$)

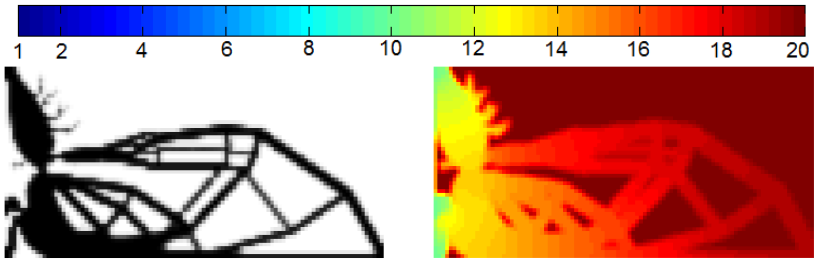
Figure 9.4 – $T_\infty = 20°\text{C}$ - TBC, ISC and BSC - Vol. frac. $= 0.5$
Nodal displacement, it.; **(a)** $u_{\text{out}} = -4.7045\mu\text{m}$, 594 (1216)



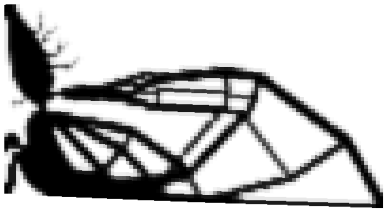
(a) Undeformed solution **(b)** Temperature distribution

(c) Deformed solution
($u_y \times 10^2$)

Figure 9.5 – $T_\infty = 1°\text{C}$ - ISC and BSC only ($h_{\min} = 10^{-3}$) - Vol. frac. $= 0.3$
Nodal displacement, it.; **(a)** $u_{\text{out}} = -2.4113\mu\text{m}$, 1149 (7126)

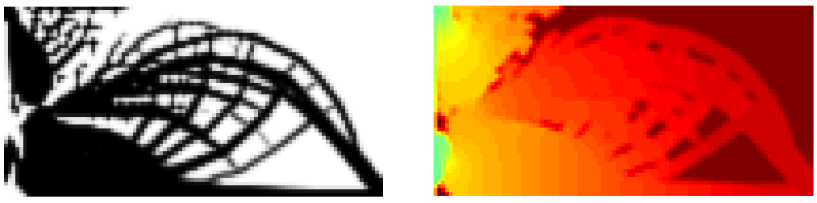


(a) Undeformed solution **(b)** Temperature distribution



(c) Deformed solution
($u_y \times 10^2$)

Figure 9.6 – $T_\infty = 20°\text{C}$ - ISC and BSC only ($h_{\min} = 10^{-3}$) - Vol. frac. $= 0.3$
Nodal displacement, it.; **(a)** $u_{\text{out}} = -3.8049\mu\text{m}$, 401 (1522)



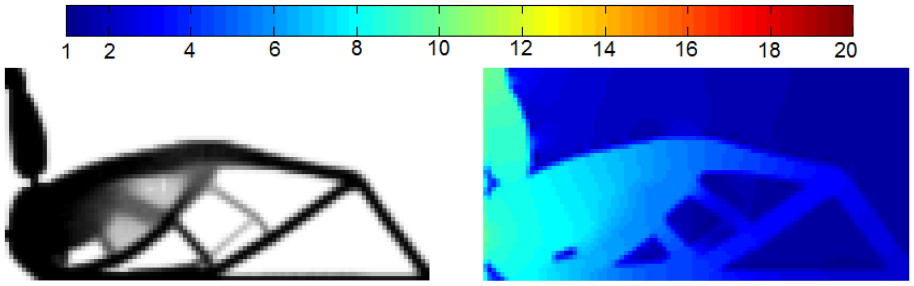
(a) Undeformed solution **(b)** Temperature distribution

(c) Deformed solution
($u_y \times 10^3$)

Figure 9.7 – $T_\infty = 20°\text{C}$ - ISC and BSC only ($h_{\min} = 10^{-3}$) - Vol. frac. $= 0.5$
Nodal displacement, it.; **(a)** $u_{\text{out}} = -4.28705\mu\text{m}$, 653 (—)

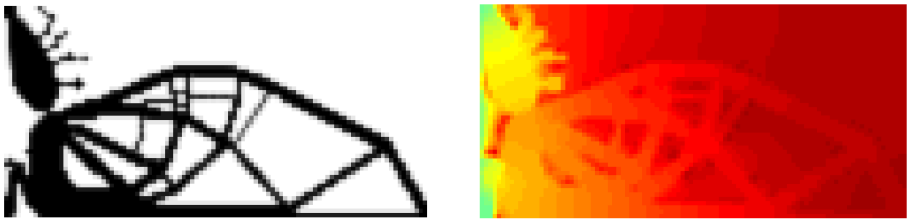
To see if the minimum convection contribution introduced in the interpolation of the ISC and BSC has any significant effect on the obtained solutions, the problem was also run with $h_{\min}$ set to 0. Although similar, substantial differences can be seen when comparing to the solutions obtained with $h_{\min} = 10^{-3}$. As figure 9.8 suggests, it appears that the solution for 1°C fluid temperature has larger areas with intermediate densities, where as the minimum convection contribution equivalent has more clearly defined boundaries. It is clearer to see the differences for the 20°C fluid temperature case, as shown in figure 9.9. There is a significant difference in the obtained solution for a volume-fraction of 0.3, the truss-like structure is no longer connected to the top actuating arm with two pivot-points, and generally looks more like the solution with TBC. For both the 30 and 50% volume cases, it appears that the "fingers" of the top actuating arm are longer than for the $h_{\min} = 10^{-3}$ counterparts. This makes sense, as they need to reach longer to reach the warmer areas of the domain.

The most important thing to notice is the differences in the temperature distributions. When the minimum convection contribution is used, the temperature outside of the structure is very close or equal to the fluid temperature as it should be. Even though the differences may not appear crucial, one cannot expect correct solutions, if the underlying physics is not modelled correctly.



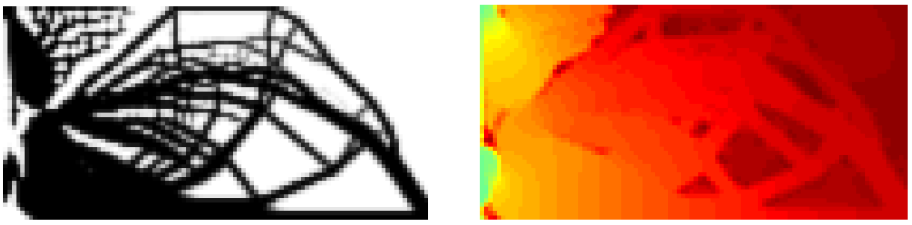
(a) Undeformed solution **(b)** Temperature distribution

Figure 9.8 – $T_\infty = 1°C$ - ISC and BSC only ($h_{\min} = 0$) - Vol. frac. $= 0.3$ Nodal displacement, it.; **(a)** $u_{\text{out}} = -2.5562\mu\text{m}$, 1269 (4445)



(a) Undeformed solution **(b)** Temperature distribution

Figure 9.9 – $T_\infty = 20°C$ - ISC and BSC only ($h_{\min} = 0$) - Vol. frac. $= 0.3$ Nodal displacement, it.; **(a)** $u_{\text{out}} = -3.8130\mu\text{m}$, 419 (978)



(a) Undeformed solution **(b)** Temperature distribution

Figure 9.10 – $T_\infty = 20°C$ - ISC and BSC only ($h_{\min} = 0$) - Vol. frac. $= 0.5$ Nodal displacement, it.; **(a)** $u_{\text{out}} = -4.1035\mu\text{m}$, 453 (1838)

Chapter 10

Conclusion

A topology optimization programme has been implemented from scratch, with the ability to optimize elastostatic mechanical problems, steady-state thermal problems and coupled thermomechanical problems, for a range of objective functions subjected to a volume-fraction constraint. The possibility to implement several other constraint functions is present, and relatively straight forward when using adjoint sensitivity analysis. The implemented features were tested and confirmed by comparison to earlier works and a somewhat equivalent MATLAB-code.

Furthermore, a design-dependent convection formulation has been implemented and tested for a variety of problems. Firstly, the FE implementation was tested on an analytical problem, several issues were outlined and suggestions on how to solve these were made. Secondly, a few test problems were solved using the expanded topology optimization features and the results were discussed.

It is concluded that the interpolation of the convection into the design-domain were successful. The results are promising and show that the design-dependent effects of convection should most definitely not be ignored. Although, this investigation has only been done for plane 2D problems, the implemented formulation should be relatively easily extended to 3D and axisymmetry. Here all sides of the elements will either be in contact with other elements or the boundary. Therefore, the effects of side convection become very important, as there is no longer a "top or bottom" as such.

Although the ultimate goal of this project was not reached; optimizing an industrially relevant problem, the project is still concluded to have been a great success. For the author especially, as this project has served as an excellent way of getting into the field of topology optimization and has been extremely giving.

10.1 Future work

Relevant paths to take for future works include the implementation of design-dependent TBC, as suggested in section 4.3.3. It is seen as an evident problem, that the convection is still applied on 0-density elements, as they have no 'physical' surface area upon the convection can act. The optimization process should, therefore, be affected by the fact that when 0-density elements are present, they are not affected by the convection of the surrounding fluid.

Further investigation into the effect of the various parameters of the ISC should also be made, hereunder the effect of the ϵ-parameter in the interpolation function in section 4.3.1 and the minimum convection contribution fraction, $h_{\min}$. It is also suggested, like Bruns [21] also suggested, that a penalisation factor could be imposed on the interpolated convection coefficient analogously to the SIMP method. This could aid in obtaining 0-1 solutions, which has been indicated to be problematic when dealing with thermal problems. Thereafter, the ISC formulation can relatively easily be extended to work for 3D and axisymmetric problems, using the same methodology.

The problems solved during this project have also been of a quite simple nature, in that they are only constrained by a limit on the amount of usable material. To fully understand the effects and implications of design-dependent convection, a wider range of more complex problems would be helpful. This would also make clear the effects of design-dependent convection on more realistic problems and in real-life situations. To achieve even greater accuracy in modelling the true nature of things, it is also possible to extend the optimization to simulations where the fluid dynamics of the surrounding fluid are taken into account. To achieve the very best and realistic results, one would need to model the fluids interaction with the shape and surface of the subject, as the efficiency of the convection is highly dependent on the fluid flow. This is, of course, an entirely different and much more complex project.

• • •

Part IV

Appendix

Appendix A

FEM and Top.Opt. programme

The programme used during this project, has been written in the FORTRAN90 programming language, using the Silverfrost Plato IDE . The majority of the FE subroutines were written by the author and partner in the course '41525 FEM-Heavy - Programming the Finite Element Method' during the period from the 2nd of September to the 2nd of December 2009. This includes elastostatic mechanical analysis, as well as steady-state thermal and coupled analysis. The program uses 4-node, 1st order, bilinear, isoparametric elements and has been implemented and tested for regular, as well as for non-regular, meshes. The input-files are based on the ANSYS-format, so that meshes easily can be exported from ANSYS and converted to be used for the program. A copy of the reports documenting the program and the implementing features can be requested by contacting the author at s072713@student.dtu.dk.

During this project, the programme has been extended to having the ability to solve mechanical and thermal optimization problems using two different solution methods. Furtherly, coupled thermomechanical optimization using the GCMMA algorithm has been implemented for different objective functions. Filtering has been implemented, where the neighbourhood of all elements are found and stored at the beginning of the optimization. All of the design sensitivities have been derived and implemented. Design-dependent convection formulations have been implemented for SC, to be used with topology optimization. The sensitivities of the various convection contributions have been derived from scratch and implemented. The ability to load a specified material distribution for use with reanalysis has also been implemented. Most of the options and parameters used in the solvers can be specified in the input-files to allow for easy parametric investigations. The source-code has been included on the attached DVD as a pdf-file and a compiled executable file.

Appendix B

Coupled sensitivity analysis

B.1 For static mechanical compliance

$$\begin{aligned} \min \quad & \Phi = \mathbf{f}^T \mathbf{u} \\ \text{s.t.} \quad & \mathbf{K}_t(\boldsymbol{\rho})\, \mathbf{t} = \mathbf{f}_t \\ & \mathbf{K}(\boldsymbol{\rho})\, \mathbf{u} = \mathbf{f}(\boldsymbol{\rho}, \mathbf{t}) \\ & \rho_{\min} \leq \rho_e \leq \rho_{\max}\,,\ e = 1, \ldots, n_e \end{aligned} \tag{B.1}$$

Two null terms are subtracted from the objective function:

$$\begin{aligned} \Phi &= \mathbf{f}^T \mathbf{u} - \boldsymbol{\lambda}_1{}^T (\mathbf{K}_t \mathbf{t} - \mathbf{f}_t) - \boldsymbol{\lambda}_2{}^T (\mathbf{K}\mathbf{u} - \mathbf{f}) \\ &= \mathbf{f}^T \mathbf{u} - \boldsymbol{\lambda}_1{}^T \mathbf{K}_t \mathbf{t} + \boldsymbol{\lambda}_1{}^T \mathbf{f}_t - \boldsymbol{\lambda}_2{}^T \mathbf{K}\mathbf{u} + \boldsymbol{\lambda}_2{}^T \mathbf{f} \end{aligned} \tag{B.2}$$

where $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ are the solutions for the adjoint problems defined later. The sensitivity of the objective function, is found by differentiating the objective function wrt. a design change:

$$\begin{aligned} \frac{\partial \Phi}{\partial \rho_e} &= \frac{\partial \mathbf{f}^T}{\partial \rho_e}\mathbf{u} + \mathbf{f}^T \frac{\partial \mathbf{u}}{\partial \rho_e} + \frac{\partial \mathbf{f}}{\partial \mathbf{t}}^T \frac{\partial \mathbf{t}}{\partial \rho_e}\mathbf{u} - \boldsymbol{\lambda}_1{}^T \frac{\partial \mathbf{K}_t}{\partial \rho_e}\mathbf{t} - \boldsymbol{\lambda}_1{}^T \mathbf{K}_t \frac{\partial \mathbf{t}}{\partial \rho_e} \\ &\quad + \boldsymbol{\lambda}_1{}^T \frac{\partial \mathbf{f}_t}{\partial \rho_e} - \boldsymbol{\lambda}_2{}^T \frac{\partial \mathbf{K}}{\partial \rho_e}\mathbf{u} - \boldsymbol{\lambda}_2{}^T \mathbf{K} \frac{\partial \mathbf{u}}{\partial \rho_e} + \boldsymbol{\lambda}_2{}^T \frac{\partial \mathbf{f}}{\partial \rho_e} + \boldsymbol{\lambda}_2{}^T \frac{\partial \mathbf{f}}{\partial \mathbf{t}} \frac{\partial \mathbf{t}}{\partial \rho_e} \\ &= \left(\mathbf{f}^T - \boldsymbol{\lambda}_2{}^T \mathbf{K}\right) \frac{\partial \mathbf{u}}{\partial \rho_e} + \left(\frac{\partial \mathbf{f}}{\partial \mathbf{t}}^T \mathbf{u} - \boldsymbol{\lambda}_1{}^T \mathbf{K}_t + \boldsymbol{\lambda}_2{}^T \frac{\partial \mathbf{f}}{\partial \mathbf{t}} \right) \frac{\partial \mathbf{t}}{\partial \rho_e} \\ &\quad + \boldsymbol{\lambda}_1{}^T \left(\frac{\partial \mathbf{f}_t}{\partial \rho_e} - \frac{\partial \mathbf{K}_t}{\partial \rho_e}\mathbf{t} \right) + \boldsymbol{\lambda}_2{}^T \left(\frac{\partial \mathbf{f}}{\partial \rho_e} - \frac{\partial \mathbf{K}}{\partial \rho_e}\mathbf{u} \right) + \frac{\partial \mathbf{f}}{\partial \rho_e}^T \mathbf{u} \end{aligned} \tag{B.3}$$

To avoid having to find the direct derivatives of the field variables, the following two adjoint problems are solved for $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$:

$$\mathbf{K}\boldsymbol{\lambda}_2 = \mathbf{f} \tag{B.4}$$

$$\mathbf{K}_t \boldsymbol{\lambda}_1 = \frac{\partial \mathbf{f}}{\partial \mathbf{t}}^T (\boldsymbol{\lambda}_2 + \mathbf{u}) \tag{B.5}$$

It is easily seen that equation B.4 is the same as the original static mechanical equilibrium equation, so:

$$(B.4) \Longrightarrow \qquad \boldsymbol{\lambda}_2 = \mathbf{u} \tag{B.6}$$

$$(B.5 - B.6) \Longrightarrow \qquad \mathbf{K}_t \boldsymbol{\lambda}_1 = 2\frac{\partial \mathbf{f}}{\partial \mathbf{t}}^T \mathbf{u} \tag{B.7}$$

With $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ as solutions to the adjoint problems above, the expression for the sensitivity can be simplified to:

$$\begin{aligned}(B.7) \Longrightarrow \frac{\partial \Phi}{\partial \rho_e} &= \boldsymbol{\lambda}_1{}^T \left(\frac{\partial \mathbf{f}_t}{\partial \rho_e} - \frac{\partial \mathbf{K}_t}{\partial \rho_e}\mathbf{t} \right) + \mathbf{u}^T \left(\frac{\partial \mathbf{f}}{\partial \rho_e} - \frac{\partial \mathbf{K}}{\partial \rho_e}\mathbf{u} \right) + \frac{\partial \mathbf{f}}{\partial \rho_e}^T \mathbf{u} \\ &= \boldsymbol{\lambda}_1{}^T \left(\frac{\partial \mathbf{f}_t}{\partial \rho_e} - \frac{\partial \mathbf{K}_t}{\partial \rho_e}\mathbf{t} \right) - \mathbf{u}^T \frac{\partial \mathbf{K}}{\partial \rho_e}\mathbf{u} + 2\frac{\partial \mathbf{f}}{\partial \rho_e}^T \mathbf{u} \end{aligned} \tag{B.8}$$

The sensitivities of the stiffness matrices and force vectors for standard thermal problems without convection are easily found from the element-wise finite element formulations:

$$\frac{\partial \mathbf{K}_t}{\partial \rho_e} = (1 - \rho_{\min})\, p {\rho_e}^{p-1} \mathbf{K}_{t,e} \tag{B.9}$$

$$\frac{\partial \mathbf{K}}{\partial \rho_e} = (1 - \rho_{\min})\, p {\rho_e}^{p-1} \mathbf{K}_e \tag{B.10}$$

$$\frac{\partial \mathbf{f}_t}{\partial \rho_e} = 0 \tag{B.11}$$

$$\frac{\partial \mathbf{f}}{\partial \rho_e} = \frac{\partial \mathbf{A}}{\partial \rho_e}\mathbf{t} = (1 - \rho_{\min})\, p {\rho_e}^{p-1} \mathbf{A}_e \mathbf{t}_e \tag{B.12}$$

When convection is included in the thermal problem, the individual element convection contributions (as stated in section 4.3) are simply added to the above expressions for $\frac{\partial \mathbf{K}_t}{\partial \rho_e}$ and $\frac{\partial \mathbf{f}_t}{\partial \rho_e}$ due to the fact that $\mathbf{K}_t = \mathbf{K}_{t,k} + \mathbf{K}_{t,h}$ and $\mathbf{f}_t = \mathbf{f}_{t,k} + \mathbf{f}_{t,h}$.

Appendix C

Analytical temperature distributions

C.1 Heat beam 1a - Top and bottom convection

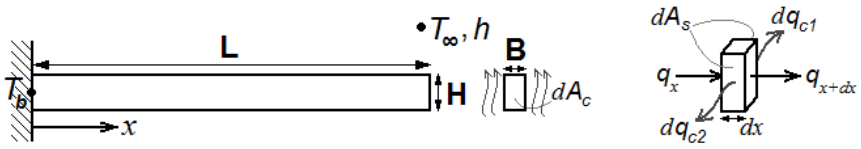


Figure C.1 – The setup for subproblem 1a.

Assumptions made: $B \ll H$ for plane conditions
$A_c = \text{const.}$
$B, H, A_s = \text{const.}$

An energy balance in the x-direction gives:

$$q_x = dq_{c1} + dq_{c2} + q_{x+dx} \tag{C.1}$$

where the individual contributions can be expressed as follows:

$$q_x = -kA_c \frac{dT}{dx} \tag{C.2}$$

$$\begin{aligned} q_{x+dx} &= q_x + \frac{dq_x}{dx} dx \\ &= -kA_c \frac{dT}{dx} + \frac{d}{dx}\left(-kA_c \frac{dT}{dx}\right) dx \\ &= -kA_c \left(\frac{dT}{dx} + \frac{d^2T}{dx^2} dx\right) \end{aligned} \tag{C.3}$$

$$\begin{aligned} dq_c &= dq_{c1} + dq_{c2} \\ &= h\, dA_s (T(x) - T_\infty) \end{aligned} \tag{C.4}$$

Substituting equations C.2 to C.4 into equation C.1 gives:

$$
\begin{aligned}
& -kA_c\frac{dT}{dx} = h\,dA_s(T(x)-T_\infty) - kA_c\left(\frac{dT}{dx}+\frac{d^2T}{dx^2}dx\right) \\
\Updownarrow \quad & kA_c\frac{d^2T}{dx^2}dx = h\,dA_s(T(x)-T_\infty) \\
\Updownarrow \quad & \frac{d^2T}{dx^2} - \frac{h}{kA_c}\frac{dA_s}{dx}(T(x)-T_\infty) = 0 \qquad \text{(C.5)}
\end{aligned}
$$

The following transformation is used to simplify the differential equation:

$$\theta(x) = T(x) - T_\infty \qquad \text{(C.6a)}$$

$$\frac{d\theta}{dx} = \frac{dT}{dx} \qquad \text{(C.6b)}$$

Equation C.5 then becomes:

$$\frac{d^2\theta}{dx^2} - m^2\theta = 0 \qquad \text{(C.7)}$$

where

$$m^2 = \frac{h}{kA_c}\frac{dA_s}{dx} \qquad \text{(C.8)}$$

as outlined in equation 4.30.

The general solution to the above differential equation is of the form:

$$\theta(x) = c_1e^{mx} + c_2e^{-mx} \qquad \text{(C.9)}$$

The boundary conditions to the current problem is a prescribed base temperature at $x = 0$ and an isolated end at $x = L$, which leads to the following boundary conditions are used to find the coefficients in the solution:

$$\theta(0) = T_b - T_\infty = \theta_b \qquad \text{(C.10a)}$$

$$\left.\frac{d\theta}{dx}\right|_{x=L} = 0 \qquad \text{(C.10b)}$$

Equations C.9 and C.10 lead to the following two equations with two unknowns:

$$c_1 + c_2 = \theta_b \qquad \text{(C.11a)}$$

$$c_1me^{mL} - c_2me^{-mL} = 0 \qquad \text{(C.11b)}$$

Solving equations C.11 give:

$$c_1 = \frac{\theta_b}{1+e^{2mL}} \qquad \text{(C.12)}$$

$$c_2 = \theta_b \frac{e^{2mL}}{1 + e^{2mL}} \tag{C.13}$$

Substituting the solutions for the coefficients into equation C.9 gives the final solution for the temperature distribution:

$$\theta(x) = \frac{\theta_b}{1 + e^{2mL}} \left(e^{mx} + e^{m(2L-x)} \right) \tag{C.14}$$

The cross-sectional area of the beam is simply found as:

$$A_c = BH \tag{C.15}$$

and the infinitesimally small change in the area affected by convection is found as:

$$dA_s = 2H dx \tag{C.16}$$

Inserting equations C.15 and C.16 into equation C.8 gives the final expression for m:

$$m^2 = \frac{h}{kBH} \frac{2H dx}{dx} = \frac{2h}{kB} \tag{C.17}$$

C.2 Heat beam 1b - Side convection

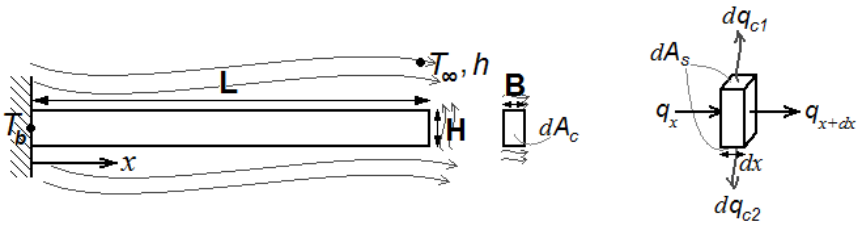


Figure C.2 – The setup for subproblem 1b.

The general solution to the differential equation is the same as for the previous problem:

$$\theta(x) = c_1 e^{mx} + c_2 e^{-mx} \tag{C.18}$$

But the boundary conditions to the current problem are different. A prescribed base temperature at $x = 0$ and an end at $x = L$ subjected to convection, which leads to the following boundary conditions are used to find the coefficients in the solution:

$$\theta(0) = T_b - T_\infty = \theta_b \tag{C.19a}$$

$$-k \frac{d\theta}{dx}\bigg|_{x=L} = h\theta(L) \tag{C.19b}$$

Equations C.18 and C.19 lead to the following two equations with two unknowns:

$$c_1 + c_2 = \theta_b \tag{C.20a}$$

$$-k\left(c_1 m e^{mL} - c_2 m e^{-mL}\right) = h\left(c_1 m e^{mL} + c_2 m e^{-mL}\right) \tag{C.20b}$$

Solving equations C.20 give:

$$c_1 = \theta_b \left(\frac{\frac{km-h}{km+h} e^{-2mL}}{1 + \frac{km-h}{km+h} e^{-2mL}} \right) \tag{C.21}$$

$$c_2 = \frac{\theta_b}{1 + \frac{km-h}{km+h} e^{-2mL}} \tag{C.22}$$

Substituting the solutions for the coefficients into equation C.18 gives the final solution for the temperature distribution:

$$\theta(x) = \frac{\theta_b}{1 + \frac{km-h}{km+h} e^{-2mL}} \left(\frac{km-h}{km+h} e^{m(x-2L)} + e^{-mx} \right) \tag{C.23}$$

The cross-sectional area of the beam is still:

$$A_c = BH \tag{C.24}$$

and the infinitesimally small change in the area affected by convection is now found as:

$$dA_s = 2Bdx \tag{C.25}$$

Inserting equations C.24 and C.25 into the equation for m gives:

$$m^2 = \frac{h}{kBH} \frac{2Bdx}{dx} = \frac{2h}{kH} \tag{C.26}$$

C.3 Heat beam 1c - Top, bottom and side convection

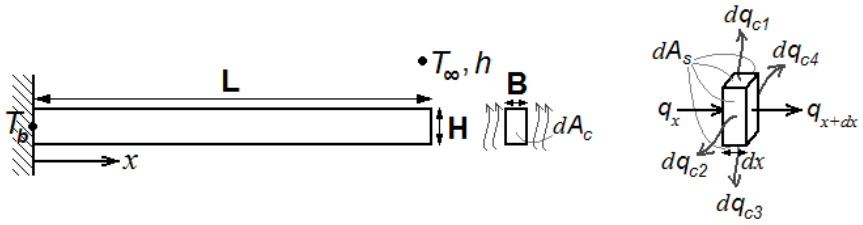


Figure C.3 – The setup for subproblem 1c.

The solution for the temperature distribution is the same as for heat beam 1b, as the boundary conditions are the same.

$$\theta(x) = \frac{\theta_b}{1 + \frac{km-h}{km+h}e^{-2mL}}\left(\frac{km-h}{km+h}e^{m(x-2L)} + e^{-mx}\right) \tag{C.27}$$

The only difference lies in the infinitesimally small change in the area affected by convection :

$$dA_s = 2(H+B)dx \tag{C.28}$$

which makes the final expression for m:

$$m^2 = \frac{h}{kBH}\frac{2(H+B)dx}{dx} = \frac{2h(H+B)}{kBH} \tag{C.29}$$

C.4 Heat beam 2

The temperature distributions for the second heat beam problems are exactly the same as for the respective subproblems for the first beam. The only difference lies in the cross-sectional area of the beam:

$$A_c = AC + BD \tag{C.30}$$

where B is defined as the height of the lower part of the T-beam, that is:

$$B = H - A \tag{C.31}$$

See figure 4.4b for definition of the other dimensions.

The infinitesimally small changes in the area affected by convection, which depends on the applied convection:

$$\text{2a:} \quad dA_s = 2(A+B)dx \tag{C.32}$$
$$\text{2b:} \quad dA_s = (C+D)dx \tag{C.33}$$
$$\text{2c:} \quad dA_s = (2(A+B)+C+D)\,dx \tag{C.34}$$

This leads to the expression for m to be as follows:

$$\text{2a:} \quad m^2 = \frac{2h(A+B)}{k(AC+BD)}dx \tag{C.35}$$
$$\text{2b:} \quad m^2 = \frac{h(C+D)}{kBH}dx \tag{C.36}$$
$$\text{2c:} \quad m^2 = \frac{h(2(A+B)+C+D)}{k(AC+BD)}dx \tag{C.37}$$

Bibliography

[1] MP Bendsøe, N Kikuchi: Generating optimal topologies in structural design using a homogenization method, Comp. Meth. Appl. Mech. Eng., 71 (1988) 197-224

[2] MP Bendsøe: Optimal shape design as a material distribution problem, Struc. Opti. (Springer), 1 (1989) 193-202

[3] N Kikuchi, K Suzuki: A homogenization method for shape and topology optimization, Comp. Meth. Appl. Mech. Eng., 93 (1991) 291-318

[4] MP Bendsøe, O Sigmund: Material interpolation schemes in toplogy optimization, Arch. Appl. Mech., 69 (1999) 635-654

[5] O Sigmund: On the Design of Compliant Mechanisms Using Topology Optimization, Mech. Struct. & Mach., 25 (1997) 493-524

[6] O Sigmund: Design of multiphysics actuators using topology optimization - Part I: One-material structures, Comp. Meth. Appl. Mech. Eng., 190 (2001) 6577-6604

[7] H Rodrigues, P Fernandes: A material based model for topology optimization of thermoelastic structures, Int. J. Numer. Methods Engng., 38 (1995) 1951-1965

[8] S Cho, JY Choi: Efficient topology optimization of thermo-elasticity problems using coupled field adjoint sensitivity analysis method , Fin. Ele. in Analy. and Design, 41 (2005) 1481-1495

[9] B Chen, L Tong: Thermomechanically coupled sensitivity analysis and design optimization of functionally graded materials , Comput. Methods Appl. Mech. Engrg., 194 (2005) 1891-1911

[10] O Sigmund: A 99 line topology optimization code written in Matlab, Struct. Multidisc. Optim., 21 (2001) 120–127

[11] E Andreassen, A Clausen, M Schevenels, BS Lazarov, O Sigmund: Efficient topology optimization in MATLAB using 88 lines of code, Struct. Multidisc. Optim., 43 (2011) 1-16

[12] O Sigmund: Morphology-based black and white filters for topology optimization, Struct. Multidisc. Optim., 33 (2007) 401–424

[13] K Svanberg: The method of moving asymptotes-a new method for structural optimization, Int. J. Numer. Methods Engng., 24 (1987) 359–373

[14] K Svanberg: A class of globally convergent optimization methods based on conservative convex separable approximations, SIAM Journ. on Optim., 14-2 (2002) 555-573

[15] K Svanberg: Notes on; MMA and GCMMA, versions September 2007, www.math.kth.se/ krille/gcmma07.pdf

[16] M Stolpe, K Svanberg: On the trajectories of penalization methods for topology optimization, Struct. Multidisc. Optim., 21 (2001) 128-139

[17] O Sigmund, J Petersson: Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima, Struct. Optim., 16 (1998) 68-75

[18] GH Yoon, YY Kim: The element connectivity parameterization formulation for the topology design optimization of multiphysics systems, Int. J. Numer. Methods Engng., 64 (2005) 1649–1677

[19] GH Yoon: Topological design of heat dissipating structure with forced convective heat transfer, J. Mech. Sci. and Tech., 24 (2010) 1225–1233

[20] A Iga, S Nishiwaki, K Izui, M Yoshimura: Topology optimization for thermal conductors considering design-dependent effects, including heat conduction and convection, Int. J. Heat and Mass Transfer, 52 (2009) 2721–2732

[21] TE Bruns: Topology optimization of convection-dominated, steady-state heat transfer problems, Int. J. Heat and Mass Transfer, 50 (2007) 2859–2873

[22] GK Ananthasuresh, L Yin: A novel topology design scheme for the multi-physics problems of electro-thermally actuated compliant micromechanisms, Sensors and Actuators A., 97–98 (2002) 599-609

[23] EJ Haug, KK Choi, V Komkov: Design Sensitivity Analysis of Structural Systems, Chapter 2, Academic Press (1986) ISBN: 0-123-32920-5

[24] MP Bendsøe, O Sigmund: Topology Optimization - Theory, Methods and Applications, Springer-Verlag (2002) ISBN: 3-540-42992-1

[25] RD Cook, DS Malkus, ME Plesha, RJ Witt: Concepts and Applications of Finite Element Analysis, Chapters 3, 6 and 12, Wiley (2001) ISBN: 978-0-471-35605-9

[26] FP Incropera, DP DeWitt, TL Bergman, AS Lavine: Introduction to Heat Transfer, Chapters 3, 4 and 6, Wiley (2006) ISBN: 978-0-471-45727-5

[27] L Eldén, L Wittmeyer-Koch, HB Nielsen: Introduction to Numerical Computation, Chapters 2 and 3, Studentlitteratur (2004) ISBN: 91-44-03727-9